ISSN 0242-9799

15 PROGRAMMES

TRACE

TIREZ PLUS DE VOTRE TRS-80* ET DE VOTRE POQUETTE SHARP

NUMERO 8

N° 8 - 1er trimestre 1984 Canada : 6,95 $ - Belgique : 230 FB - Suisse : 12 FS 30 F

SYBEX

Gagnez un Voyage à Silicon Valley

Oui, vous êtes invités gratuitement à gagner un voyage d'une semaine pour deux personnes au pays de la micro-informatique.

MICRO-EXPO, 9e congrès-exposition, carrefour international de la micro-informatique se tiendra à Paris, au Palais des Congrès du 22 au 26 mai 1984.

Visitez cette manifestation qui vous offrira la possibilité exceptionnelle de rencontrer et de dialoguer avec plus de 200 exposants français et étrangers, de suivre une trentaine de conférences professionnelles et grand public : comment choisir son tableur électronique, les systèmes intégrés : 1 - 2 - 3, Lisa, Visi/On, MS-WIN, choisir son micro, comptabilité et bases de données, Basic...Découvrez les dernières nouveautés dont certaines seront présentées en exclusivité.

La multiplicité et la diversité des produits et techniques présentés à ce grand rendez-vous annuel constitueront pour vous la garantie du bon investissement et de la bonne décision.

COUPON RÉPONSE A COMPLÉTER ET A RETOURNER A SYBEX - 6-8, impasse du Curé - 75018 PARIS

Celui-ci est votre titre de participation au tirage au sort qui aura lieu en présence de Maître P. Chale, huissier à Paris. Merci de m'adresser :

☐ une entrée gratuite et le programme détaillé des conférences.

☐ un passeport valeur 100 F T.T.C. me donnant droit à l'entrée permanente au salon, au guide de la micro 84 et de participer à toutes les conférences de mon choix (attention le nombre de places est limité !).

NOM ....

Prénom ....

Société ....

N° .... Rue ....

Code postal .... Ville ....

Activité de l'entreprise ....

Fonction ....

Ci-joint chèque de 100 F.

MICRO EXPO

TRACE

N° 8

Sommaire

Editorial

Sauve qui peut ! Les ordinateurs envahissent la planète ! Ce pourrait être le message du dernier jeu à la mode (il existe probablement déjà) apparaissant sur l'écran vidéo de votre ordinateur... C'est pourtant, et plus que jamais, une réalité quotidienne. Le mouvement amorcé depuis quelques années ne cesse de se confirmer. "L'année de l'ordinateur", c'était hier. Il va falloir se "creuser les méninges" pour trouver un autre slogan qui qualifiera cette nouvelle année 1984, car en ce domaine, elle s'annonce déjà pleine de promesses. Les nouveaux ordinateurs se multiplient, de même que les manifestations dans lesquelles ils apparaissent.On ne s'étonne plus aujourd'hui, de voir un ordinateur par exemple sur les stands de la foire agricole !

A l'époque où l'ordinateur individuel commençait à entrer dans les foyers, les professionnels le contemplaient d'un air narquois : comment pourraient-ils sérieusement employer de tels "jouets" dans leur travail... Pourtant, certains ont eu l'idée de le faire et ils ne le regrettent pas aujourd'hui. Ils ont même une certaine avance sur leurs confrères qui s'aperçoivent - mais un peu tard - qu'une entreprise qui n'est pas informatisée à l'heure actuelle, ne pourra pas subsister.

Certains constructeurs, comme Texas et Tandy, ont suivi le mouvement et se tournent désormais et résolument - non sans avoir donné le coup de grâce à leur "home computer" respectif - vers les applications professionnelles, du moins si l'on observe les campagnes publicitaires concernant les matériels proposés : modèle 3, modèle 4, modèle 100 et bientôt le Tandy-2000 pour ce dernier constructeur. L'ordinateur à la maison n'était-il donc qu'un prétexte ?

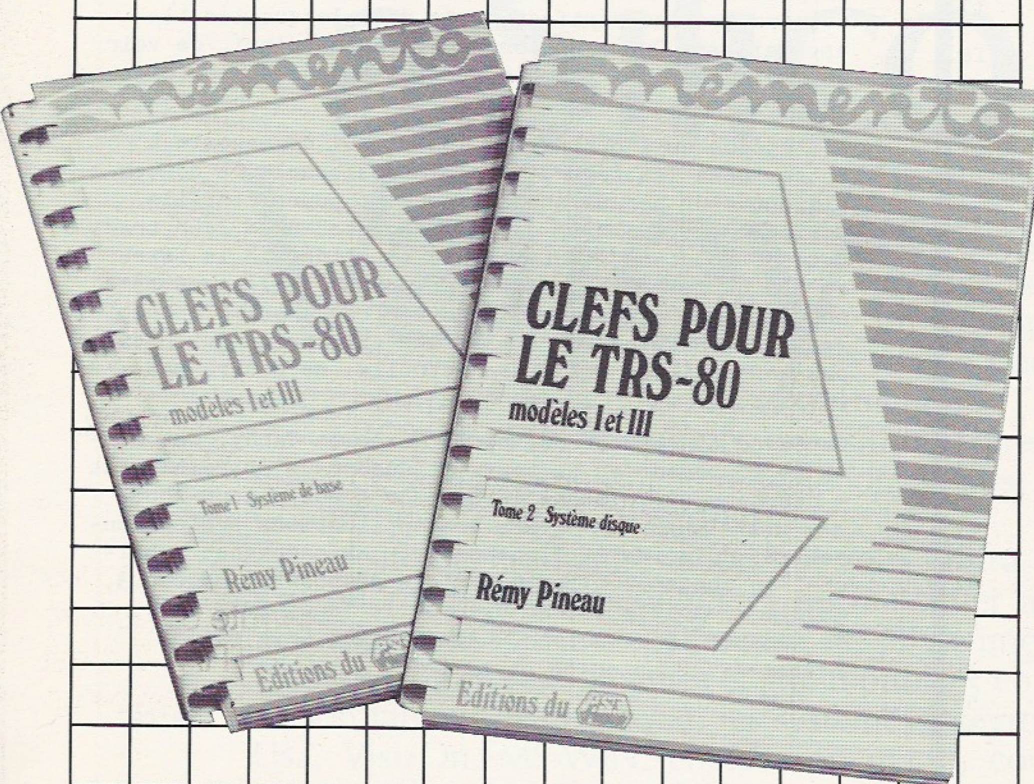
TRACE

Rédacteur en chef : Alain Pinaud - *Editeur :* Bernard Savonet - *Directeur de la publication :* Jean-Pierre Nizard - *Maquette :* Studio Agraph - *Secrétariat :* Nicole Aleman.
Rédaction et abonnements : Editrace, 8, rue Saint-Marc, 75002 Paris - *Régie publicitaire :* Force 7, Anne Jourdan, 39, rue de la Grangé-aux-Belles, 75484 Paris Cedex 10 - Tél. : 238.66.10 - *Diffusion auprès des boutiques informatiques et des librairies :* Edition du PSI, BP 86 - 77402 Lagny-sur-Marne Cedex.

Abonnements 4 numéros (un an) : France 105 FF ; Etranger 120 FF ; par avion 170 FF.

Ont collaboré à ce numéro : Kaarina Alain, Pascal Abrivard, Philippe Arques, Bernard Besse, Camille Bochard, Josué Bonifas, Roger Brousmiche, Roger Buvat, Michaël Dubois, Philippe Esclangon, Daniel Glazman, Gérard Grandpierre, L. Joly, Antoine Joux, Noël Lhomme, D. Petit, Rémy Pineau, Stéphane Poupard, Elie Ranchoux, Serge Rostan, Christophe Schlick, Christian Trigueros, Yannick Wautelet.

LES INDISPENSABLES



Clefs pour le TRS-80 - *Tome 1*
par Rémy Pineau

Débutant ou spécialiste, ce "mémento" vous est destiné. Son but est de vous fournir les "clefs" d'un accès rapide et efficace à l'apprentissage ou à la maîtrise de l'utilisation des Modèles I et III. Ce premier tome aborde essentiellement les caractéristiques de la version de base du TRS-80 (plans, mémoires, brochages, codes, syntaxe, message d'erreurs du Basic, codes machine Z80); cependant les deux derniers chapitres, consacrés aux adresses ROM et RAM du système, servent de transition au 2[e] tome destiné aux utilisateurs actuels ou potentiels de disque.

192 pages - 120,00 FF/925 FB/37,60 FS

Clefs pour le TRS-80 - *Tome 2*
par Rémy Pineau

Possesseurs de TRS-80 modèle I ou III équipé de disques, ce deuxième tome des "clefs" vous est destiné. Il a pour objectif de vous donner les principales informations concernant l'utilisation des disques à travers l'étude des différents formats de stockage de données, des commandes des Systèmes d'Exploitation Standards (TRS-DOS, LDOS, NEWDOS) et des utilitaires associés les plus courants.

200 pages - 120,00 FF/925 FB/37,60 FS



P.S.I. DIFFUSION
BP 86 - 77402 Lagny-S/Marne Cedex
FRANCE
Téléphone (6) 006.44.35

P.S.I. BENELUX
5, avenue de la Ferme Rose
1180 Bruxelles
BELGIQUE
Téléphone (2) 345.08.50

P.S.I. SUISSE
Case Postale
Route Neuve 1
1701 FRIBOURG
SUISSE
Téléphone (037) 23.18.28
CCP 17.56.84

au Canada
SCE Inc.
65, avenue Hillside
Montréal (Westmount)
Québec H3Z1W1
Tél. : (514) 935.13.14

Envoyer ce bon accompagné de votre règlement à P.S.I. DIFFUSION ou, pour la Belgique et le Luxembourg à P.S.I. BENELUX ou, pour la Suisse à P.S.I. SUISSE.

DESIGNATION	NOMBRE	PRIX
par avion : ajouter 8 FF (75 FB) par livre	TOTAL	

Signature (obligatoire pour paiement par carte de crédit)

☐ Paiement par chèque joint ☐ Paiement en FF par carte bleue VISA (à P.S.I. DIFFUSION uniquement)



N° ____ Date d'expiration ____

NOM ____ PRENOM ____

rue ____ n° ____

Code postal ____ Ville ____



Nous commençons à "crouler" sous le courrier et les articles que vous nous envoyez... ce qui tend à prouver l'intérêt que vous portez à notre revue, à VOTRE revue. Merci d'avoir répondu à nos appels !

Certains lecteurs s'étonnent de ne pas voir passer leurs articles plus rapidement ou nous disent : "j'espère que cet article passera dans le prochain numéro ...". Que ces derniers se rassurent : ils n'ont pas été oubliés et leur tour viendra : La parution d'un article est soumise à différentes contraintes : la "file d'attente" des articles reçus antérieurement (et elle commence à être importante), la longueur de l'article (un article long nous cause toujours beaucoup de problèmes, car compte tenu du nombre de pages du magazine, il ne peut occuper une place trop importante, sauf s'il présente un intérêt pour la grande majorité de nos lecteurs, ce dont nous ne sommes jamais sûrs...), le sujet dont traite l'article (un sujet inédit ou d'actualité, a des chances de passer plus rapidement qu'un classique "jeu du pendu" ou qu'un programme utilitaire qui risque de ne toucher que 2 % de nos lecteurs). De plus, il y a l'occasion ou, pour diverses raisons, il paraît souhaitable qu'un article soit publié conjointement à un autre, ce dernier se trouvant alors extrait de la pile avant terme. Enfin, certains articles, bien que présentant un grand intérêt, ne peuvent pas être publiés "tels quels et demandent à être remaniés, complétés, ou appuyés par des exemples (qui devront naturellement être testés au préalable). Compte tenu du temps dont nous disposons, il est évident que ces articles, indépendament de notre volonté, risquent de redescendre la file d'attente plutôt que de la remonter : Soyez donc patients ... et compréhensifs. Merci.

If not possible ?

- Dans le livre "Visicalc sur TRS", je n'arrive pas à entrer la formule de la case A20 pour l'exemple de la feuille d'impôt : après la frappe de "@IF", le fait d'entrer le signe "(" fait disparaître "@IF"... (M. Dupuis, Roost).

* La raison est très simple : la fonction IF est inconnue de la version de Visicalc tournant sur TRS-80. En entrant la parenthèse le programme détecte alors qu'il ne connaît pas cette fonction et l'efface. Le problème vient du fait que le livre en question a initialement été écrit sur Apple, ordinateur pour lequel Visicalc est plus riche et possède notamment la fonction IF.

Le complexe du TRS !

- Mon TRS-80 modèle 1 doit faire un complexe d'infériorité quand il s'agit de faire tourner les programmes : "conduite de nuit", "un utilitaire qui vaut de l'or" et "apprenez la musique"... Au fait, testez-vous les programmes qui vous sont envoyés ? Cela rejoint le problème général des programmes paraissant dans les revues : à de rares exceptions, des modifications sont nécessaires pour les faire tourner (Raymont Nivot, Le Creusot).

* Tous les programmes que nous publions sont testés sur un TRS-80 modèle 1 et cela, pour trois raisons : pour vérifier les quelques modifications que nous apportons, pour calculer les sommes de contrôle de chaque ligne (afin d'éviter à nos lecteurs de faire des erreurs de frappe qui risqueraient justement de donner des complexes à leur ordinateur) et enfin, pour en sortir la liste sur imprimante. Naturellement, nous ne pouvons pas garantir un fonctionnement correct du programme dans tous les cas de figure (nous avons aussi d'autres choses

à faire et nous sommes obligés de faire un peu confiance à l'auteur !) mais nous vérifions toujours que le programme se lance correctement et s'exécute sans incident après quelques minutes de fonctionnement. De même, il ne nous est pas possible de vérifier la bonne éxécution du programme sur chaque type ou chaque version d'ordinateur autre que celui utilisé pour nos tests. Comme nous sommes faillibles (enfin, presque pas !), nous avons prévu une rubrique spéciale dans notre magazine (Step-1), afin de réparer d'éventuelles erreurs. A ce jour (et pourtant, nous en recevons du courrier !), aucune erreur ne nous a été signalée qui compromette le fonctionnement de l'un des programmes dont il est question. Au fait, avez-vous réellement envisagé la possibilité d'une erreur qui ne viendrait ni de nous, ni de votre ordinateur...?

TRS la honte !

- J'ai essayé d'adapter le programme TRS-80 "pour mieux prévoir" (pages 47 à 49, Trace n° 2) au ZX81 et j'ai été surpris de constater des différences importantes au niveau du calcul du coefficient de corrélation délivré par ces deux machines, ce qui fausse le choix du meilleur ajustement. Pris d'un doute, j'ai repris le programme sur ma TI-58C et ai constaté qu'elle donnait raison au ZX ! (Jean-Michel Petit, Magny-Cours).

  * Merci pour ces "précisions" sur... l'imprécision du BASIC TRS, ce qui incite à conclure que dans ce cas, on a sûrement intérêt à travailler en double précision.

Deux "token" pour ELSE

- Suite à la parution du programme "Plus-Quot" pour produire des "super-graphiques (Trace n° 4, page 52), j'ai quelques difficultés avec le mot-clé ELSE qui produit les deux codes : 149 et 58, ce qui donne des graphiques bizarres (Jean Suchocki, Vrigne aux Bois).

  * La commande ELSE est en effet un cas particulier (de même que l'apostrophe utilisée pour les commentaires) qui produit deux "token", le second étant le séparateur de commande ":" (code 58). Une solution peu satisfaisante consiste à décaler le graphique d'une colonne ou d'une ligne pour ne pas tomber sur le code graphique 149... Un lecteur a peut-être trouvé une meilleure solution ?

Y'a plus qu'à !

- Du sérieux, SVP ! Avant de prendre position au sujet du modem (Carte MDX-2, Trace n° 6). Il faut placer les bornes 13 et 14 du circuit intégré au 0 volt et calculer les nouveaux filtres. (L. Loriaux, Sambreville).

  * Nous aurions "sérieusement" accepté que vous nous donniez ces nouvelles valeurs de filtres ! Mais pour vous montrer que finalement, nous prenons les choses au sérieux, nous envisageons de publier bientôt un article sur ces modifications.

Appel de fonction illégal ?

- "Illegal fonction call" : quelle erreur embêtante ! Je connais le BASIC, mais pas au point de savoir que faire quand elle arrive. Pourriez-vous m'aider ? (J-M Dubasik, Biesset).

  * Cette erreur apparaît lorsque vous tentez d'appeler une fonction alors que les paramètres d'appel sont incorrects, par exemple : racine carrée d'une valeur négative, appel USR vers une routine en langage machine dont l'adresse n'est pas spécifiée (ou incorrectement spécifiée), etc. Lorsque cette erreur arrive, imprimez (PRINT) le paramètre en question et vous devriez comprendre rapidement la cause de cette erreur. Exemple :

```
10 N=3
20 PRINT SQR(N)
30 N=N-1
40 GOTO 20
```

  Lorsque l'erreur arrive, faire :

```
? N
```

  La valeur imprimée est "-1", ce qui confirme bien l'erreur affichée.

Ne pas confondre avec la gaufrette du patissier...

- Je ne suis pas d'accord avec votre traduction du mot "wafer" par "galette" dont l'équivalent français serait plutôt "gaufrette" (cf Dictionnaire Français-Anglais Larousse, mais 1960, page 724), d'autant plus que le mot "galette" peut prêter à confusion avec les galettes de bandes magnétiques utilisées sur les platines de magnétophone, et qui, à ma connaissance, n'ont jamais été utilisées en informatique (Gérard Barré, Beauvais).

  * Merci pour cette précision, mais ne cherchez pas trop dans les dictionnaires anglais la signification américaine de certains mots... Vous risquez parfois d'en être très étonné : Les Anglais eux-mêmes ne sont pas toujours d'accord de l'usage que font les Américains de leur vocabulaire !

MAGAZINE

Tandy est maintenant implanté en Europe depuis 10 ans où il compte à ce jour, 515 magasins dont 112 en France et 194 en Belgique. L'usine de production des TRS-80, MTE (Matra-Tandy-Electronique), produit aujourd'hui pour l'ensemble du réseau européen, les modèles 3, 4 et MC-10. Plus de 20000 unités seraient déjà sorties de cette usine.

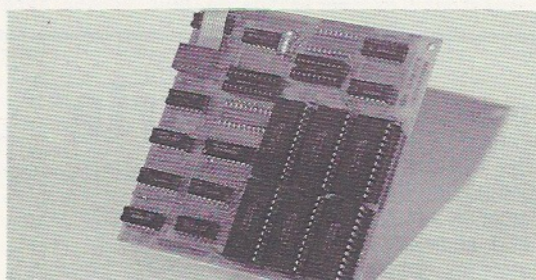
Tandy qui a complètement renouvelé sa gamme TRS80, a introduit sur le marché plus de 6 nouveaux modèles d'ordinateurs individuels en 6 mois et plus de 10 modèles d'imprimantes en 1 an. De nouveaux appareils ont été présentés récemment à l'occasion de l'exposition Comdex de Las Vegas. Ils seront disponibles en France durant le premier semestre 84 : fin mars pour le modèle 4 P (modèle 4 portable), fin juin pour le modèle 2000 (16 bits à base de microprocesseur Intel 8088, tournant sous le système d'exploitation MS-DOS - tout comme l'IBM-PC).

Notons également l'ouverture à Saint-Etienne du dixième "Centre Ordinateurs" Tandy en France, dont l'adresse est :

Rue Marc Charras
42000 Saint-Etienne
(Tél : 77/38.51.09)

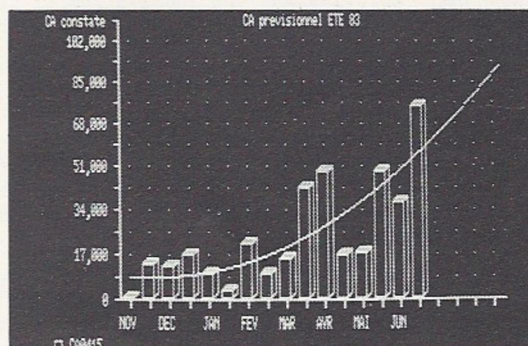
où des cours gratuits d'initiation au BASIC et au traitement de texte sont proposés sur modèle 3.

Tandy France, 211-213 bd Mac Donald, 75019 Paris.

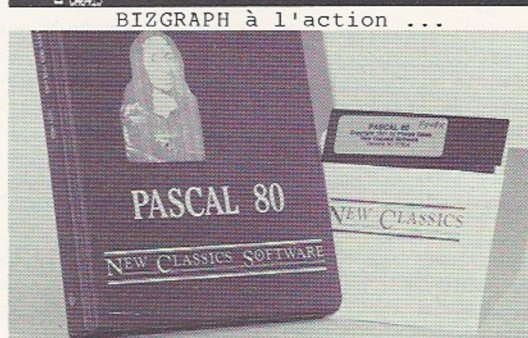


Carte Grafyx Solution

IDEN, ainsi que les sociétés Micro-Influx et Boutisoft, commercialisent les produits suivants :
- la carte haute résolution "Grafyx Solution" pour le modèle 3. S'installant sans aucune soudure, elle offre (en noir et blanc) une résolution de 512 x 192 points. La carte est accompagnée d'un manuel de 50 pages et d'une disquette (ou une cassette selon le cas) comprenant une extension du BASIC pour commander les nouvelles fonctions graphiques et plus de 20 programmes, dont une recopie d'écran pour la plupart des imprimantes graphiques connues à ce jour. Un pilote d'écran (driver) permet l'affichage de lignes de 80 caractères. Prix de vente : 3290 FF TTC.
De plus, les logiciels Bizgraph et Draw, vendus respectivement 1490 et 495 FF TTC, complètent utilement cette extension. Le premier, orienté "affaires", permet d'afficher des graphiques sous plusieurs formes : histogrammes, camemberts, courbes, etc... Quant au second programme, il s'agit d'un éditeur graphique utilisé pour la conception de dessins.
- Pascal 80 pour les modèles 1 et 3, commercialisé ici dans une nouvelle version qui inclut entre autre : les variables pointeurs, la double précision sur toutes les fonctions mathématiques, les fonctions graphiques et aléatoires, un substitut aux "variant records". Pascal-80 est livré sur disquette et est accompagné d'un manuel bien documenté. Prix de vente : 1290 FF TTC.



BIZGRAPH à l'action ...



Le Pascal 80 pour modèles 1 et 3

Autre nouvelle (et surprise) qui intéressera probablement nos lecteurs : à la suite de nombreuses demandes (dont nous ne sommes pas étrangers...), ces sociétés ont décidé de commercialiser en France le système d'exploitation de disquettes NEWDOS-80 version 2 pour les modèles 1 et 3, qui était pratiquement introuvable jusqu'alors. Une bonne initiative qui va faire des heureux !

IDEN, 34 bis rue Sorbier, 75020 Paris
Micro-Influx, 20 rue Laennec, 78330 Fontenay le Fleury
Boutisoft, 9 rue de Lalande, 33000 Bordeaux

L'AFPA (Association pour la Formation Professionnelle des Adultes) organise des stages de courte durée (du lundi au vendredi) dans les domaines suivantes : électronique générale, logique séquentielle, microinformatique, logiciel en microinformatique, téléinformatique. Ces stages sont organisés toute l'année sauf juillet et août.

AFPA, 38 ave Victor Hugo, 38800 Le Pont de Claix.

Le progiciel Compta-3 de la société Série-3, est maintenant présenté dans une nouvelle version dont les caractéristiques sont les suivantes :

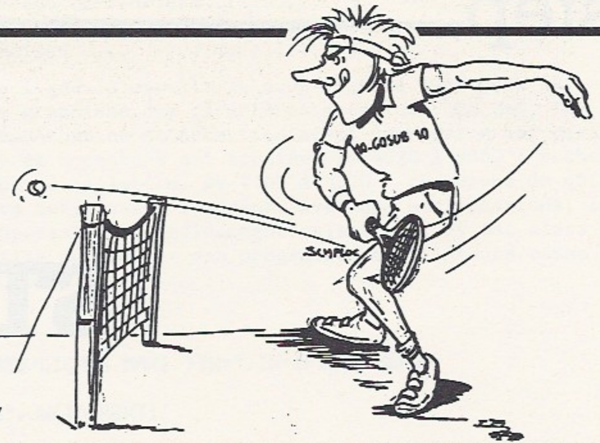
- version mixte Nouveau et Ancien Plan Comptable,
- numérotation 5 chiffres (au lieu de 4),
- nouvelle Balance Générale à plusieurs niveaux de sélectivité,
- paramètrage du bilan et CEG,
- optimisation Edition journal/imputation comptes
- nouvelle documentation en trois brochures : "Visa pour Compta-3","initiation et mise en route", "Utilisation des programmes",
- version TRS-DOS 1.3 ou LDOS,

Prix : 1750 FF TTC.

Série-3, 5 rue Mont Alaric, 11100 Narbonne.

Côté court





LISTEZ VOS MOTS-CLES EN 62 OCTETS !

Voici un programme de listage des mots-clés avec leurs "token" encore plus court que celui publié dans le numéro 6 de TRACE (62 octets au lieu de 66), à condition de supprimer les espaces...

```
10 FOR I=5712 TO 6175 : IF PEEK(I) > 128 PRINT,J+128;
: J=J+1
20 PRINT CHR$(PEEK(I)AND NOT 128); : NEXT
```

G. Briand

RETROUVEZ LE NOM D'UN PROGRAMME "SYSTEM"

Ce "mini-utilitaire" est destiné exclusivement aux utilisateurs de cassettes sur modèle 1. Il permet de retrouver le nom d'un programme "system" que l'on a oublié. La routine en langage machine qu'il contient pourra être placée dans la zone réservée à l'horloge, inutilisée ici. Ainsi, il n'y a pas de risque de destruction, même après un "memory size" malencontreux.

```
10 DATA 62,0,205,18,2,205,150,2,6,7,205,
        53,2,205,51,0,16,248,195,204,6
```

Pour charger ce programme, faire :

```
FOR A=16478 TO 16498 : READ B : POKE A,B : NEXT
```

et pour l'exécuter :

```
SYSTEM
/16478
```

Vous verrez apparaître 7 lettres sur l'écran : un "U" puis les six lettres du nom recherché. Si vous ne voyez pas de "U", la bande n'est pas de type "system" ou le réglage de volume du cassetophone est incorrect.

Pierre Duchêne

Dans la série "plantez votre système"...

Tapez :

```
SYSTEM
/133
```

puis frappez plusieurs touches en observant bien l'écran...

Pour obtenir des variantes, on peut aussi essayer les adresses 183, 1414, et 1415.

(Il reste encore 12 284 possibilités de branchement dans la MEM... Tentez votre chance ! Ndlr).

Yves De Herdt

TROIS PETITS PROGRAMMES POUR JOUER AVEC DE GRANDS NOMBRES

Les trois programmes qui suivent comportent chacun 3 lignes de BASIC. L'un calcule les grandes factorielles d'un nombre, l'autre les grandes puissances, quant au dernier, il donne le quotient d'une division décimale connaissant le dividende (DVD) et le diviseur (DVS), avec un certain nombre de chiffre après la virgule... Chacun d'eux est suivi d'un exemple d'exécution.

```
0 '                * * *   FACTORIELLES   * * *

1 DEFINT J : L=30 : DIM N(L+1) : N(0)=1 : P=1E3 : PRINT N"! =";
: REM 83
2 N=N+1 :FOR J=0 TO L :IF N(J)=)P,Q=INT(N(J)/P) :N(J)=N(J)-Q*P
: N(J+1)=N(J+1)+Q :NEXT ELSE NEXT:FOR J=L TO 0 STEP-1 :IF N(J)=0
: NEXT ELSE IF J=L,END : REM 51
3 FOR J=J TO 0 STEP-1 : PRINT" "RIGHT$(STR$(P+N(J)),3); : NEXT
: PRINT :FOR J=0 TO L :N(J)=N(J)*N :NEXT :PRINT N"! ="; :GOTO 2
: REM 6D

 63 ! = 001 982 608 315 404 440 064 116 146 708 361 898 137 544
773 690 227 268 628 106 279 599 612 729 753 600 000 000 000 000
 64 ! = 126 886 932 185 884 164 103 433 389 335 161 480 802 865
516 174 545 192 198 801 894 375 214 704 230 400 000 000 000 000
```

```
0 '                * * *   PUISSANCES   * * *

1 DEFINTJ :L=30 :DIM N(L+1) :M=1 :P=1E3 :INPUT N :N(0)=N:GOSUB 3
: REM E1
2 PRINT:FOR J=0 TO L:N(J)=N(J)*N:NEXT:GOSUB 3:IF N(J))0 RUN ELSE
: FOR J=L TO 0 STEP-1:IF N(J)=0,NEXT ELSE M=M+1:PRINT N"↑"M" =",
: FOR J=J TO 0 STEP-1:PRINT" "RIGHT$(STR$(P+N(J)),3);:NEXT:GOTO2
: REM 1D
3 FOR K=0 TO L : IF N(K)=)P THEN Q=INT(N(K)/P) : N(K)=N(K)-P*Q
: N(K+1)=N(K+1)+Q : NEXT:RETURN ELSE NEXT:RETURN : REM 87

 2 ↑ 98  =        316 912 650 057 057 350 374 175 801 344
 2 ↑ 99  =        633 825 300 114 114 700 748 351 602 688
 2 ↑ 100  =       001 267 650 600 228 229 401 496 703 205 376
 2 ↑ 101  =       002 535 301 200 456 458 802 993 406 410 752
```

```
0 '                * * *   DECIMALES   * * *

1 DEFDBL N,M : INPUT"DVD";N : PRINT,CHR$(27)"/ "; : INPUT"DVS";M
: REM 3C
2 Q=INT(N/M) : N=N-Q*M : P=1E3 : PRINT,,CHR$(27)" = " Q ","
: REM 1F
3 Q=INT(N*P/M) :N=N*P-Q*M :PRINT RIGHT$(STR$(Q+P),3)" "; :GOTO 3
: REM 5E

RUN
DVD? 314159265  / DVS? 271828183 =  1 ,
155 727 347 815 145 422 209 587 443 697 844 972 903 343 138 632
538 333 966 643 922 274 976 174 931 795 059 675 618 697 712 444
334 736 255 070 358 175 480 281 233 384 840 011 236 068 189 441
563 533 535 446 543 451 309 461 903 734 978 061 491 144 205 602
```

Jean Mevel

STEP -1

Cette rubrique est destinée (comme son nom l'indique aux programmeurs BASIC que vous êtes !) à apporter des précisions, des améliorations ou à corriger des erreurs concernant les articles ou programmes des numéros précédents. Elle est née et sera alimentée en grande partie par vos lettres. Aussi, ne vous en privez pas ! Merci.

COURRIER DES LECTEURS / TRACE n° 7, page 10

DISQUETTES TRACE SUR MODELE 3 AVEC NEWDOS80 V2.
Nous avons oublié de spécifier un point important pour relire une disquette Trace sur modèle 3 avec NEWDOS-80 V2 : la commande WRDIRP qui permet de réécrire la directory de la disquette selon les spécifications du contrôleur floppy employé sur le modèle 3 (différent de celui du modèle 1). Nous résumons donc ci-dessous les différentes manipulations à effectuer en partant du principe que NEWDOS-80 V2 est sur l'unité 0 et la disquette Trace sur l'unité 1 (les 2 disquettes doivent être déprotégées en écriture) :

```
PDRIVE,0,1,TI=A,TD=A,TC=35,SPT=10,TRS=3,
     GPL=2,DDSL=17,DDGA=2,A
WRDIRP,1
```

Après exécution de ces deux commandes, toutes les opérations (DIR, COPY, etc.) deviennent possibles sur la disquette Trace.

COTE COURT / TRACE n° 6, page 16

Le programme de listage des mots-clés comporte une "coquille". Il faut lire "IF 127 > P"...

TRAITEMENT DE TEXTE / TRACE n° 5 et 6

Equipé d'un TRS modèle 3 non sonorisé, je propose l'une des solutions suivantes pour remplacer le "bip" de fin de ligne.

- Une règle de longueur JL, graduée de 5 en 5, à conserver sur 2 lignes "non scroll" en haut d'écran. Ajouter alors la ligne :

```
1125 CLS : POKE 16916,2 : PRINT STRING$(JL,45) : FOR
K=0 TO JL+1 STEP 5 : PRINT TAB(K) CHR$(33); : NEXT
```

et débuter la ligne 1130 par "PRINT : JP=..."

- Un signal optique (un # par exemple), apparaissant 5 caractères avant la fin de ligne. En ce cas, conserver :

```
50 IF N=JL-5 GOSUB 630
```

et remplacer la ligne 630 par :

```
630 PRINT TAB(N) CHR$(35); : RETURN
```

Noël Lhomme

PROGRAMME BASPY / TRACE n° 6

Avec les modifications suivantes, l'utilitaire Baspy peut accepter les nombres décimaux dans les commandes D, E et R, si ils sont précédés de l'identificateur "+". Il suffit de remplacer la ligne 4220 par :

```
04220  HEXBIN  LD   A,(HL)
04221          INC  HL
04222          CP   '+'
04223          JR   Z,DECBIN
04224          DEC  HL
04225          LD   DE,OFFFFH
```

puis, pour initialiser le programme, de remplacer : SYSTEM */31534 par : SYSTEM */31541.

Antoine Joux

TRUCS A BRAC / TRACE n° 6, page 17

Les mots de passe "partout" pour TRS-DOS 1.2 sur modèle 3 étaient erronés et sont à remplacer par :

OTIS , PNEUW , RATPD

Dans les versions 1.3 ou 1.3I, TRS-DOS ne permet plus (le malin !) d'utiliser un mot de passe universel.

RECOPIE DE CASSETTES SUR MODELE 3 / TRACE n° 4

Pour fonctionner sur ce modèle (à 500 bauds seulement), le programme est à modifier comme suit :

```
- ligne 130 DATA 205,232,49   : REM CALL 31E8H
- ligne 140 DATA 243          : REM DI
```

Roger Buvat

TRAITEMENT DE TEXTE (ENCORE LUI !) TRACE n° 5 et 6

Les modifications qui suivent sont destinées à permettre au programme de traitement de texte d'imprimer des caractères accentués et des lignes de titre en caractères dilatés sur l'imprimante Seikosha GP-80M. Dans le texte, les caractères accentués seront repérés par leur lettre suivie d'un chiffre, selon la codification suivante :

Accent aigu 1
Accent grave 2
Accent circonflexe 3
Tréma 4
Cédille 5

La ligne 12 appelle la sous-routine 5900 pour définir le graphisme des caractères accentués. En 360, la sous-routine de détection avant impression des caractères accentués est appelée. De 4020 à 4060 : recherche de la lettre. De 4100 à 4520 : recherche du chiffre définissant l'accent. Avec ces modifications, l'impression est évidemment ralentie... Il est alors utile de trouver des programmes compilateurs comme Accel 3.

```
 11 DIMJW(20)
 12 GOSUB5900
334 INPUT"NOMBRE DE LIGNES TITRES < 40 LETTRES/LIGNES MAXI >IW=";IW:IFIW=0THEN34
0
335 FORI=0TOIW-1:INPUT "N# DES LIGNES TITRES";JW(I):NEXTI
350 FORI=1TOM:IFK1=1LPRINTUSING"###:";I;
355 FOR J=0TOIW-1:IFI=JW(J)THENKW=1
356 NEXTJ
357 IFKW=1THEN365
360 LPRINTTAB(MA);CHR$(15);:GOSUB3000:LPRINTCHR$(15):GOTO366
365 LPRINTTAB(MA);CHR$(14);S(I):LPRINTCHR$(15)
366 KW=0:NEXTI
3000 L%=LEN(S(I))-1
3002 FORJ=1TOL%+1
3005 W$=MID$(S(I),J,1):W%=VAL(MID$(S(I),J+1,1))
3010 IFW%=0THEN3015ELSE3016
3015 LPRINTW$;:GOTO3020
3016 GOSUB4020
3017 LPRINTCHR$(15);
3020 NEXTJ
3030 RETURN
4020 IFW$=CHR$(97)THEN4100
4025 IFW$=CHR$(99)THEN4150
4030 IFW$=CHR$(101)THEN4200
4040 IFW$=CHR$(105)THEN4300
4050 IFW$=CHR$(111)THEN4400
4060 IFW$=CHR$(117)THEN4500
4070 LPRINTW$;:RETURN
4100 IFW%=2THENLPRINTA2$;:J=J+1:RETURN
4110 IFW%=3THENLPRINTA3$;:J=J+1:RETURN
4150 IFW%=5THENLPRINTC7$;:J=J+1:RETURN
4200 IFW%=1THENLPRINTE1$;:J=J+1:RETURN
4210 IFW%=2THENLPRINTE2$;:J=J+1:RETURN
4220 IFW%=3THENLPRINTE3$;:J=J+1:RETURN
4230 IFW%=4THENLPRINTE4$;:J=J+1:RETURN
4300 IFW%=3THENLPRINTI3$;:J=J+1:RETURN
4310 IFW%=4THENLPRINTI4$;:J=J+1:RETURN
4400 IFW%=3THENLPRINTO3$;:J=J+1:RETURN
4500 IFW%=2THENLPRINTU2$;:J=J+1:RETURN
4510 IFW%=3THENLPRINTU3$;:J=J+1:RETURN
5900 A2$=CHR$(8)+CHR$(128)+CHR$(160)+CHR$(213)+CHR$(214)+CHR$(188)+CHR$(192)+CHR
$(128):REMA ACCENT GRAVE
5910 A3$=CHR$(8)+CHR$(128)+CHR$(160)+CHR$(214)+CHR$(213)+CHR$(190)+CHR$(192)+CHR
$(128):REMA ACCENT CIRCONFLEXE
5930 E1$=CHR$(8)+CHR$(128)+CHR$(184)+CHR$(212)+CHR$(214)+CHR$(213)+CHR$(200)+CHR
$(128):REM  E ACCENT AIGU
5940 E2$=CHR$(8)+CHR$(128)+CHR$(184)+CHR$(213)+CHR$(214)+CHR$(212)+CHR$(200)+CHR
$(128):REM E ACCENT GRAVE
5950 E3$=CHR$(8)+CHR$(128)+CHR$(184)+CHR$(214)+CHR$(213)+CHR$(214)+CHR$(200)+CHR
$(128):REM E ACCENT CIRCONFLEXE
5960 E4$=CHR$(8)+CHR$(128)+CHR$(184)+CHR$(213)+CHR$(214)+CHR$(213)+CHR$(200)+CHR
$(128):REM E TREMA
5970 I3$=CHR$(8)+CHR$(128)+CHR$(198)+CHR$(253)+CHR$(194)+CHR$(128):REM I ACCENT
CIRCONFLEXE
5980 I4$=CHR$(8)+CHR$(128)+CHR$(197)+CHR$(252)+CHR$(193)+CHR$(128):REM I TREMA
6000 O3$=CHR$(8)+CHR$(128)+CHR$(188)+CHR$(198)+CHR$(197)+CHR$(198)+CHR$(188)+CHR
$(128):REM O ACCENT CIRCONFLEXE
6100 U2$=CHR$(8)+CHR$(128)+CHR$(188)+CHR$(193)+CHR$(194)+CHR$(188)+CHR$(192)+CHR
$(128):REM U ACCENT GRAVE
6110 U3$=CHR$(8)+CHR$(128)+CHR$(188)+CHR$(194)+CHR$(193)+CHR$(190)+CHR$(192)+CHR
$(128):REM U ACCENT CIRCONFLEXE
6200 C7$=CHR$(8)+CHR$(128)+CHR$(152)+CHR$(164)+CHR$(228)+CHR$(180)+CHR$(164)+CHR
$(128):REM C CEDILLE
7000 RETURN
```

Philippe Arques

Trucs à Brac

TRANSFERT DE CONVERT ET DE XFERSYS SUR MODELE 3 AZERTY

Comme cela a déjà été signalé, ces 2 utilitaires ne sont pas toujours fournis sur la disquette TRS-DOS version 1.3I correspondant au clavier AZERTY. Pourtant, CONVERT est nécessaire pour lire une disquette modèle 1, telle que celle de TRACE par exemple, en raison des différences de densité et d'organisation du catalogue (directory). Ces programmes étant protégés, voici une méthode permettant de réaliser leur transfert, même avec une seule unité, et aussi bien sur le modèle QWERTY que sur l'AZERTY.

- Sous Disk BASIC, lancer le programme suivant :

```
10 FOR I=(&H8040) TO (&H804B) : READ J : POKE I,J :
   NEXT
20 DATA 33,0,82,17,0,112,1,64,16,237,176,201
```

- Tapez ensuite :

CMD "S"

puis :

DUMP TOTO/CMD (START=8040,END=804B,TRA=8040)

- Introduire la disquette TRS-DOS version 1.3 (pour clavier QWERTY) et taper :

LOAD CONVERT/CMD

- Placer ensuite la disquette contenant le "relocateur" TOTO/CMD et lancer ce programme :

TOTO

- Monter alors la disquette TRS-DOS version 1.3I (pour clavier AZERTY et taper :

DUMP CONVERT/CMD (START=7000,END=803F,TRA=5200)

puis enfin :

RELO CONVERT/CMD (ADD=5200)

On pourra procéder de la même façon pour XFERSYS, les adresses étant identiques, avant de "remercier" TOTO en le supprimant de la disquette...

Roger Buvat

TAPEZ DANS VOS MAINS !

Sur TRS modèle 1 et dans le cas d'un programme comme par exemple "casse briques" publié dans le numéro 6 de TRACE, on peut commander le départ de la balle par un bruit... plutôt que par l'appui sur une touche, ce qui a toujours pour effet d'étonner les invités ! Il suffit, dans le cas du programme cité plus haut, de remplacer la ligne 300 par :

```
300 IF INP(255) > 127 THEN NC=NC+1 : GOSUB 390
```

Ouvrir alors le couvercle du cassetophone, presser l'ergot situé au font à gauche et appuyer simultanément sur les touches RECORD et PLAY. Penser aussi à retirer la fiche AUX.

Christophe Schlick

ASSEMBLEZ SUR STRINGY FLOPPY ESF

Après avoir (laborieusement) chargé Edtasm+ à partir de la cassette, vous l'avez bien sûr, enregistrée sur "wafer" par la commande "∂SAVE1,17280,12093,17280". Par la même occasion, vous avez aussi enregistré "Stand Alone Z-Bug" par : "∂SAVE2, 17280,6125,17280".
Voilà déjà un progrès considérable qui vous permet de charger votre assembleur favori en quelques secondes. Mais cela ne suffit pas : il faudrait aussi pouvoir sauver les programmes source sur "wafer"...
Pour ce faire, il suffit de savoir que le programme source débute à l'adresse 29217 (25854 si Z-Bug a été "quashé") et que la première adresse disponible après le programme source (valeur du "first" obtenue sous éditeur par la commande *Q) est contenue dans les mémoires 4232H et 4233H (poids faibles, poids forts), soit : 16946 et 16947. La longueur L du programme source est donnée par "first" en décimal moins 29217 (moins 25854 si Z-Bug "quashé"). Pour sauver sur wafer un programme source, il faut donc :

- revenir au BASIC (commande *B)
- réactiver l'ESF (SYSTEM /12345)
- faire : ∂SAVE(n),16946,2,6681
- faire : ∂SAVE(n+1),29217,L,17283

Le retour en 6681 (1A19H) redonne le contrôle au BASIC lors du chargement. Si Z-Bug a été "quashé", la seconde commande SAVE devient : SAVE(n+1),25854,L, 17283.
Pour charger un programme source sous EDTASM+ (après chargement de EDTASM), retour au BASIC par *B et réactivation de l'ESF), il suffit de faire : ∂LOAD(n) et ∂LOAD(n+1).
La sauvegarde sur wafer d'un programme assemblé "inmemory" se fait de la façon classique : ∂SAVE(n),adr1,lg,adr2 ou adr1 est l'adresse origine du programme (ORG), lg sa longueur en octets, et adr2 son adresse de lancement.

L. Joly

Le TRS-80 modèle 4



TRSDOS 06.01.00F- Copyright 1983 Logical Systems Inc.
All Rights Reserved. Licensed to Tandy Corporation.
TRSDOS61 08/15/83

Le TRS80 modèle 4 est un ordinateur monobloc semblable au modèle 3, comprenant un clavier complet AZERTY avec 2 couches inédites (CTRL et CAPS), un "pad" numérique (touches 0 à 9, point décimal, touche ENTER) et trois touches de fonctions (F1, F2, F3). Un écran vidéo vert haute résolution de 30 cm de diagonale lui permet d'afficher soit 16 lignes de 64 caractères soit 24 lignes de 80 caractères. Un emplacement est prévu pour recevoir deux minidisquettes 5 pouces 1/4 avec leur alimentation. Reprenant sous une couleur blanche la carrosserie du modèle 3, il possède comme lui un bouton "RESET" encastré à côté du clavier numérique, des commandes de luminosité et de contraste (sous le bord gauche du clavier) et un interrupteur de tension (sous le bord droit du clavier).
Le TRS80 modèle 4 est disponible selon trois configurations :

- 16 k / Cassette (fonctionnement sous mode Modèle 3)
- 64 k / 1 disquette 184 k / TRSDOS version 6 (horloge 4 MHZ)
- 64 k / 2 disquettes 184 k / TRSDOS version 6 (horloge 4 MHZ)

et peut se voir ajouter :

- une imprimante parallèle
- deux disquettes extérieures
- une interface RS232
- une carte haute résolution (153600 points, 240 x 640)
- un disque dur
- 64 k de mémoire RAM (portant le maximum à 128 k RAM)

Sous l'appareil sont disposées des ouvertures recevant les câbles de liaison avec les périphériques : imprimante parallèle, minidisquettes additionnelles, RS232, interface BUS (pour la carte graphique par exemple). La prise DIN pour le cassettophone est située à côté de la prise d'alimentation secteur, seule prise reliant l'ordinateur au courant domestique.

Sur son simple aspect extérieur, le modèle 4 semble donc n'être qu'un modèle 3 superficiellement modifié, une sorte de "Modèle 3e"...!! En réalité, il n'en est rien comme nous allons le voir plus loin.

En ouvrant le carton contenant le matériel, vous trouverez (en France) :

- Un modèle 4 avec clavier AZERTY
- Un manuel (en anglais !) du Système d'Exploitation de Disquettes
- Une carte de référence BASIC & SED (en anglais !)
- Un manuel (en anglais !) d'initiation au BASIC des modèles I/III (!!!)
- Un livret (en anglais !) de présentation pédagogique (?) du SED et d'un programme d'application en BASIC (MAILLIST)
- Une minidisquette SF/DD 40p. contenant le SED.

C'est tout de même assez maigre et inadapté, d'autant plus que les manuels sont érronés et/ou incomplets et que manque toujours le manuel technique, indispensable pour une programmation des entrées/sorties ou en assembleur.
Tandy assure que les traductions sont en cours mais demeure assez discret sur le suivi du SED et la disponibilité du Technical Reference Manual (pourtant très bien conçu !!! je l'ai lu !!!).

Le SED et la documentation du modèle 3 ne sont pas fournis avec le modèle 4 et donc, seuls ceux possédant déjà ces "matériaux" pourront bénéficier de la totale compatibilité entre ces deux systèmes.
Initialisé avec un SED modèle 3, le TRS80 modèle 4 se comporte exactement comme un "vrai" modèle 3 disposant de ses 14 k de ROM et de ses 48 k de RAM (horloge 2 MHZ). En fait, il y a quelques changements dans les ROM comme le montre très bien le programme (MEMTEST) du TRSDOS 1.3 mais ces modifications sont justement celles qui permettent au modèle 4 d'être, au choix, un modèle 3 ou un modèle 4.

A la mise sous tension le TRS80 -4 est en fait un modèle 3 dont seule la ROM C a été modifiée pour autoriser l'initialisation vidéo. Lors du démarrage du SED un "disk bootstrap loader" lit le secteur 1 de la piste 0 en RAM ; ce secteur contient un second "bootstrap loader" (BOOT/SYS) et ainsi une disquette modèle 3 s'initialise parfaitement. Pour le TRSDOS version 6, BOOT/SYS comprend un chargeur complémentaire qui initialise les 64 k RAM, réalise les modifications vidéo, et "déconnecte" les ROM modèle 3.

Tous les tests de disquettes modèle 3 sur modèle 4 en mode modèle 3 que j'ai pu effectuer se sont révélés positifs, y compris pour des programmes comme SU+, Visicalc,...etc. Par contre, sous mode modèle 4, c'est-à-dire après chargement en RAM du TRSDOS version 6, les programmes modèle 3 ne sont plus acceptés, du fait des différences importantes entre les BASIC, de l'architecture même du modèle 4 et de l'environnement vidéo.

En mode 4, le SED comme le BASIC sont entièrement en RAM. On ne peut plus effectuer les classiques PEEK et POKE pour l'écran ou le clavier ni appeler des routines en ROM !!!

Le BASIC Microsoft 01.00.00 qui n'est pas sans rappeler celui de l'IBM/PC, comprend de nouvelles instructions mais, dans le même temps, ne supporte plus certains mots-clés présents dans le BASIC modèle 3.

MOTS CLES NON SUPPORTES

RESET SET CMD CLOAD
CSAVE POINT

MOTS CLES SUPPORTES DIFFEREMMENT

SYSTEM RESTORE CLEAR PRINT @
VARPTR PRINT TAB THEN

En attendant l'extension graphique haute résolution (640 x 240) et son BASIC graphique, les commandes SET et RESET n'existent plus et leur suppression n'est pas précisée dans le manuel du modèle 4. La nouvelle commande PRINT @ (ligne, colonne) permet toutefois de "simuler" ou palier l'absence de SET/RESET :

Ex : Trace d'un bord d'écran

```
 1 ' demo
 5 CLS
10 ?@(5,10),"DEMO INSTRUCTION (PRINT@(LIGNE,COLONNE))"
15 ?@(7,5) ,"LISTING :"
17 ?@(9,5) ,"FOR I = 0 TO 79 :@? (0,I),CHR$(191);:?
   @(22,I),CHR$(191);:NEXT"
18 ?@(10,5),"FOR I= 0 TO 22 : ?@(I,0),CHR$(191);:?@
   (I,79),CHR$(191);:NEXT"
20 FOR I = 0 TO 79 : ?@(0,I),CHR$(191);:?@(22,I),CHR$
   (191);:NEXT
30 FOR I = 0 TO 22 : ?@(I,0),CHR$(191);:?@(I,79),CHR$
   (191);:NEXT
40 '
50 ?@(23,0),"TAPEZ (BREAK) POUR ARRETER ...";
60 GOTO 60
```

La suppression des commandes CLOAD et CSAVE tient au fait que le modèle 4 sous TRSDOS version 6 ne supporte pas les entrées/sorties cassette, sauf via l'utilitaire TAPE100/CMD et pour la liaison avec le TRS modèle 100. De là une redéfinition complète de la commande SYSTEM qui vient remplacer la commande CMD.

MOTS CLES NOUVEAUX OU SUPPORTES DIFFEREMMENT

CALL	CHAIN	CLEAR	COMMON	DATE$	ERASE	ERR$
HEX$	INPUT$	LPOS	OCT$	OPTION	PRINT@	PRINT TAB
ROW	SPACE$	SPC	SWAP	SOUND	SYSTEM	VARPTR#
WAIT	WHILE/WEND	WIDTH	WRITE	WRITE#		

CALL : Permet d'appeler un programme en langage machine et de lui transmettre des paramètres.
CHAIN : Permet de chaîner des programmes BASIC avec diverses options de passage de variables ou de fusion.
CLEAR : Le M4 effectue une allocation dynamique de l'espace mémoire et CLEAR n'intervient alors que pour définir une adresse mémoire et l'espace réservé par rapport à cette adresse.
COMMON : Permet le passage de variables lors d'un CHAIN ou MERGE.
DATE$: Retourne la date d'initialisation.
ERASE : "Efface" les variables spécifiées.
ERR$: Retourne la dernière erreur SED.
HEX$: Donne la valeur HEXA d'un nombre décimal.
INPUT$: Accepte une chaîne d'un nombre spécifié de caractères depuis le clavier ou un fichier et n'affiche pas les caractères lus.
LPOS : Retourne la position courante de la tête d'impression imprimante.
OCT$: Donne la valeur octale d'un nombre décimal.
OPTION : Définition du plus petit indice pour les données en tableau.
PRINT@ : Existe sous la forme classique PRINT@ (position) mais aussi sous la forme PRINT @(ligne,colonne) - cf listing.
PRINT TAB : Idem M3 mais exploite les chiffres supérieurs à 79.
ROW : Retourne la position ligne courante du curseur.
SPACE$: Génère une chaîne d'un nombre spécifié d'espaces.
SPC : Génère des espaces.
SWAP : Echange des contenus de 2 variables.
SYSTEM : Remplace CMD. Permet sous BASIC d'appeler des commandes SED.
VARPTR : Retourne la valeur adresse mémoire d'un buffer.
WAIT : Suspension de l'exécution d'un programme.
WHILE/WEND : Effectue une série d'opérations dans une boucle.
WRITE : Ecriture sur l'écran.
WRITE # : Ecriture dans un fichier séquentiel.
SOUND : Non documentée, cette instruction permet de générer, sans adjonction d'un amplificateur, un son de tonalité et de durée programmables. Ex : SOUND1,1 (la hauteur de la note est définie par un chiffre entre 0 et 7 et la durée par un chiffre entre 0 et 31).

Ex : Gamme des sons

```
 1 ' demo
 5 cls
10 ?@(0,5),"Vous allez entendre les SONS produits par
   le modèle 4"
20 FOR I = 0 TO 7
30 SOUND I,0
40 NEXT
```

Si le nouveau BASIC 01.00.00 constitue l'élément le plus évident d'incompatibilité entre le M4 (sous mode M4) et le M3, c'est en fait l'architecture en RAM qui en constitue l'élément le plus essentiel et en même temps, la grande nouveauté du modèle 4.

Le TRSDOS version 6, issu du LDOS 5.1.4 et les SVC 'appels superviseurs' feront l'objet d'un prochain article.

En attendant, et pour "jouer", voici de quoi changer votre curseur clignotant et protéger contre le défilement les lignes supérieures de votre écran.

```
 0 'Modification du curseur
 1 J=0:DIM UV%(9):UV%(0)=15872:UV%(2)=8448:UV%(4)=
   4352:UV%(6)=256:UV%(8)=239:UV%(9)=201
 2 ' CETTE LIGNE 1 INITIALISE LES APPELS SVC.
 3 ' ELLE NE DOIT PAS ETRE RELANCEE
 5 INPUT "QUEL CARACTERE (1-255) ";X
10 UV%(1)=15:UV%(7)=CVI(CHR$(X)+CHR$(8))
20 GOSUB 100
30 CLS: ?@(10,10),"Voilà qui est fait... (BREAK) pour
   arrêter ";: GOTO 5
100 DEF USR0 = VARPTR (UV%(0)): J%=USR0(0): RETURN
```

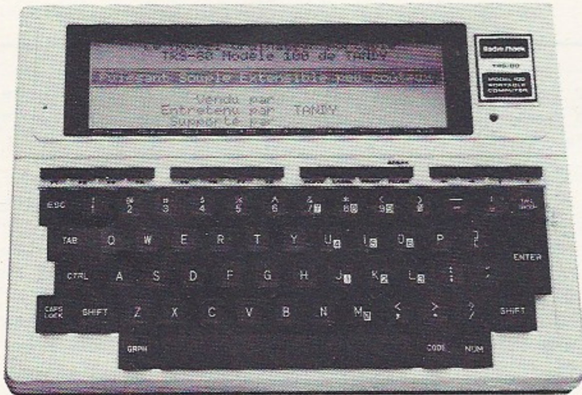
```
 0 'Protection contre le défilement
 4 ' Reprendre les lignes 1 et 100 du programme
   précédent
 5 INPUT "Combien de lignes ( 0 - 7 ) ";X
10 UV%(1)=15:UV%(7)=CVI(CHR$(X)+CHR$(7))
20 GOSUB 100
30 CLS: ?@(X+1,5),"Voilà qui est fait...";
35 FOR I = 1 TO 100 :? "Pour supprimer cette protec-
   tion répondez (0) à la demande ":next"
40 Goto 5
100 'Gosub avec DEFUSR1 ou 2
```

Enfin pour les curieux du clavier AZERTY qui comprendraient mal l'utilisation du préfixe 0 dans l'affichage des codes graphiques voici de quoi réaliser cet affichage :

```
10 'Affichage des codes du TRS à tester ensuite avec
11 'les commutateurs CHR$(21) et CHR$(22)
20 dim a(255)
30 for i = 1 to 255
40 a$(i) = chr$(0)+chr$(i)
50 ? ";=";i;a$(i),;:next
60 goto 60
```

Serge Rostan

Et si on parlait un peu du modèle 100?



Le Modèle 100 portable de Tandy est une petite machine révolutionnaire à plus d'un titre. L'habitude de l'écran magique incorporé (ou non) aux ordinateurs individuels fait souvent considérer avec une moue dédaigneuse cette petite "calculette" qui ne paie vraiment pas de mine. Que les vrais initiés gardent pour eux ce secret : le modèle 100 démode complètement les vieilles guimbardes aux fils qui traînent partout, à la mémoire défaillante et à l'encombrement qui fait pousser les hauts cris aux maîtresses de maison soucieuses du bon ordonnancement de leur cher foyer !

Car enfin, il n'est rien que ne puisse faire le modèle 100 ! Avec 24 K octets il dépasse en performances tout ce que l'on peut imaginer, et n'a aucun rapport avec les "poquettes", si modernes soient-elles... Et son écran de 40 x 8 (320 signes), ou encore ses 239 x 63 (15057 pixels adressables individuellement), remplace avantageusement les écrans des modèles de table. (Quand avez-vous eu besoin pour la dernière fois, de plus de 8 lignes de 40 signes sur votre écran ?).

Travailler dans le train ou sur sa table de jardin, puis rentrer chez soi pour établir un listage ou des étiquettes-adresses, sans être rivé à un poste fixe, cela fait repenser le problème informatique. Le matériel qui vous suit partout ne vous coupe plus de votre vie de famille, et les performances sont aussi bonnes qu'avec un modèle cinq fois plus gros...

DES AVANTAGES INCOMPARABLES POUR LES HABITUES DE L'ORDINATEUR

LES PROGRAMMES INCORPORES. Le programme de traitement de texte, bien qu'il soit rudimentaire, a l'immense avantage d'être immédiatement prêt à l'emploi dès que vous mettez en route votre machine. Simple, rapide et performant, il vous épargnera dans bien des cas l'achat d'un Supercripsit, et amortira le coût du modèle 100. On peut d'ailleurs améliorer cet éditeur et l'adapter au genre de machine que l'on possède.

LES PROGRAMMES EN MEMOIRE VIVE. Eteignez votre modèle 100. Reprenez-le le lendemain : vos programmes sont encore là, exécutables dans la seconde qui suit... Quel progrès par rapport à ce que vous avez connu avec votre vieux modèle 1 : charger la cassette, changer de programme, recharger la cassette, etc... Ici, plus de problème : vous accumulez en mémoire deux, cinq, dix programmes simultanés : seule, la capacité de la mémoire vous arrêtera.
Mieux encore, lorsque vous concevez un programme, il vous arrive de tâtonner puis d'y revenir le lendemain pour y apporter une dernière retouche. A chaque fois, que ce soit sur disquette ou sur cassette, vous devez répéter les opérations rituelles (CSAVE, CLOAD? ou SAVE "BIDULE/BAS:1) pour chaque retouche. Maintenant, plus de ces opérations fastidieuses : votre programme est en permanence en mémoire, et ce n'est qu'une fois que vous aurez définitivement terminé votre mise au point que vous pourrez passer à l'enregistrement sur cassette. De plus, pendant toute la durée de la mise au point, vous n'aurez pas eu à craindre la panne de courant ou les facéties de l'électricité statique qui venaient autrefois "planter" vos programmes.

QUELQUES FONCTIONS INTERESSANTES. IPL vous permet, en allumant l'ordinateur, de mettre en route un programme automatiquement. SOUND x,y fournit des petits sons sur cinq octaves. TIME$ vous permet de stopper l'ordinateur à un moment donné, ou de passer à telle ou telle opération avec l'instruction "ON TIME$ GOTO" (ou GOSUB). KILL supprime un fichier, NAME le renomme et MERGE: chaîne deux fichiers...

ET QUELQUES INCONVENIENTS FACHEUX...

- L'absence de certaines instructions bien utiles comme TRON ou TROFF (qui permettent de suivre pas à pas un programme et de savoir où l'on s'est trompé) et DELETE, si pratique pour détruire quelques numéros de lignes parasites.

- La "mémoire" des touches de la machine : si, à la question "appuyer sur enter", vous appuyez douze fois au lieu d'une, l'ordinateur gardera en mémoire les 11 appuis excédentaires et exécutera les ordres en conséquence. Au début, on est surpris...

- Le fait que les TRS modèle 100 ne contiennent pas la même mémoire morte en Europe et aux Etats-Unis peut causer de gros problèmes si l'on suit aveuglément tel ou tel conseil donné par le manuel d'instructions (notamment pour redonner aux touches de fonction leur usage original...).

- L'absence totale de logiciels. En attendant que le catalogue Tandy s'enrichisse, il ne reste plus qu'à les faire nous-mêmes...

D. Petit

Quelques PEEK et POKE pour modèles 1, 3 et compatibles

Pour répondre à la demande du "courrier des lecteurs" de TRACE n° 6, voici quelques indications sur les utilisations les plus fréquentes de PEEK et POKE appliquées au TRS80 et Compatibles (VGS).
Les utilisations ont été regroupées par types et ne prennent pas en compte les appels aux routines MEM. Sur ce point on se reportera à l'indispensable "Microsoft Basic Decoded..." de J. Favour.

. Ø125H (293) : Test pour Modèle 1 ou 3.
Si PEEK(293)=73, c'est un Modèle 3.
Le même test sur un VGS-I donne 13 et non 73.
Ce test, placé en début de programme, peut donc permettre d'aiguiller vers les instructions appropriées au même titre que les tests effectués pour déterminer si l'on a affaire à une configuration disque ou cassette. (Voir ci-dessous)

. 37E1H (14305) : Sur le Modèle 1 cette adresse permet la sélection de l'unité de disquette alors que sur le Modèle 3 on utilise le Port 244 (F4H).
Une routine assez fréquente est celle qui consiste à justement sélectionner une unité pour la laisser ensuite en rotation entre les diverses opérations d'entrées et de sorties.
Pour ce faire, on teste également le statut du contrôleur, à l'adresse 37ECH (14316) sur le Modèle 1 (via le Port 240 (FØH) sur le Modèle 3.
Cette routine est donc :

```
Modèle 1 : 1Ø POKE 14305,1 : IF PEEK (14316) < 128
           THEN 1Ø
Modèle 3 : 1Ø OUT 244, 1 : IF INP (240) < 128 THEN 1Ø
```

. 37E8H (14312) : Test du statut Imprimante. Ce test fonctionne aussi sur VGS (derniers modèles) bien que la documentation précise le Port 253. Ainsi, pour un VGS et une imprimante OKI 80:"?PEEK(14312)" et "?=INP (253)" sont équivalents et les valeurs lues sont :

Allumée + Off : 159 (9F)   Allumée + On : 63 (3F)
Eteinte + Off : 31 (1F)   Eteinte + On : 63 (3F)

L'adresse 4Ø28H (16424) est située dans le bloc de contrôle (nombre de lignes/page) et est toujours initialement positionnée à 67.

. 38ØØH (14336)
3BFFH (15359) : Ces adresses sont celles du début et de la fin de la mémoire clavier. Les tests les plus usuels sont ceux effectués via les adresses 1442Ø et 144ØØ pour les touches "flèches" :

Flèche en Haut = 8   Flèche en Bas = 16
Flèche Gauche = 32   Flèche Droite = 64

La "carte clavier" est donnée dans la Pratique du TRS 80 volume 3 (Editions du PSI). (Voir aussi dans ce même numéro, l'article sur les différences entres les claviers QWERTY et AZERTY du Modèle 3 -Ndlr).

. 3CØØH (1536Ø)
2FFFH (16383) : Ces adresses sont celles du début et de la fin de la mémoire vidéo (écran).
Le Modèle 3 dispose d'une possibilité de protection contre le scrolling : on peut protéger 8 lignes (codes Ø à 7) via l'adresse 4212H (16916).
Le mode d'affichage de l'écran est testé à l'adresse 16445 sur le Modèle 1 et 16912 sur le modèle 3 ;

	32 Caractères	64 Caractères
Modèle 1 : TEST =	Ø	8
Modèle 3 : TEST =	4Ø	44

Pour passer en affichage 32 caractères, on peut donc éviter le CHR$(23) et POKER Ø (ou 44) selon le Modèle. L'utilisation des Ports 255 (Modèle 1) et 253 (Modèle 3) s'effectue suivant le même principe mais ne semble pas fonctionner corectement sur VGS.

. 4ØØCH (16396) : Test Cassette/Disquette
Si la réponse est 2Ø1, la configuration est à cassette. Ce test a déjà été indiqué par TRACE mais le revoici :

If Peek (16396) = 2Ø1 then Poke 16526,LS :Poke 16527, MS Else Defusr = L
(LS=octet de poids faible, MS=octet de poids fort, L=adresse routine USR).

** Si les programmes intègrent systématiquement ce test ainsi que celui du Modèle il est possible de proposer des versions "multi-configurations", pourvu bien sûr que soient incluses toutes les variantes lorsque c'est nécessaire.

. 4Ø2ØH (16416)
4Ø21H (16417) : Adresse position curseur dans la mémoire écran.

. BREAK

La gestion de la touche BREAK se réalisera ainsi :

Level II : adresse 16405
Modèle 3 : adresse 16396
TRSDOS 2.3 : adresse 23886
NEWDOS 2.1 : adresse 23461

Si on POKE Ø (165 pour le Modèle 3), la touche est invalidée et le retour à la normale s'effectue en "Pokant" 2Ø1.
Pour NEWDOS 80/2.Ø l'adresse est 4312H (1717Ø) et l'invalidation s'obtient en "Pokant" ØC3H(195) et le retour en "Pokant" ØC9H(2Ø1).

Serge Rostan

Différences entre les claviers QWERTY et AZERTY du modèle 3

Deux modifications essentielles interviennent :
- 1 au niveau de la matrice clavier,
- 2 au niveau du pilote clavier en MEM.

	Adresses →							
Donnees ↓	3801	3802	3804	3808	3810	3820	3840	3880
1		H	P	X	Ø	8	ENTER	SHIFT G.
2	A	I	Q	Y	1	9	CLEAR	SHIFT D.
4	B	J	R	Z	2	*:	BREAK	
8	C	K	S		3	+;	↑	
16	D	L	T		4	<,	↓	
32	E	M	U		5	=-	←	
64	F	N	V		6	>.	→	
128	G	O	W		7	?/	SPACE	

Clavier Qwerty

	Adresses →							
Donnees ↓	3801	3802	3804	3808	3810	3820	3840	3880
1	<>	H	P	X	Ø	8 -	ENTER	SHIFT G.*
2	Q	I	A	Y	1	9	CLEAR	SHIFT D.
4	B	J	R	W	2	°)	BREAK	
8	C	K	S	¨^	3	‾_	↑	
16	D	L	T	*ȼ	4	≠$	↓	
32	E	?,	U	M	5	.;	←	
64	F	N	V	ù%	6	/:	→	
128	G	O	Z	{ }	7	+=	SPACE	

Clavier Azerty
* + clavier numérique



	1 &	2 é	3 "	4 /	5 (	6 g	7 è	8 !	9 ç	Ø à	°)	— -	# $	BREAK
Etiquettes :	! 1	" 2	# 3	$ 4	% 5	& 6	/ 7	(8	) 9	Ø	* :	+ ;	< ,	
	↑	A	Z	E	R	T	Y	U	I	O	P	¨ ^	* ȼ	←	→
Etiquettes :		Q	W											
	↓	Q	S	D	F	G	H	J	K	L	M	% ù	ENTER	
Etiquettes :		A												
	SHIFT	W	X	C	V	B	N	? ,	. ;	/ :	+ =	SHIFT		
Etiquettes :		Z						M	=	.	? /			
	< >	SPACE	}{											
Etiquettes :	ȼ													



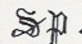
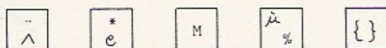


tableau 3

Les modifications intervenant au niveau du pilote clavier et de la matrice clavier sont dues :

- à la présence physique de 5 touches supplémentaires :



- à la répartition différente de ces touches sur le clavier.

Les cinq touches supplémentaires correspondent au décodage des bits 3, 4, 5, 6 et 7 de l'adresse 3808.

Clavier Azerty et pilote clavier Qwerty : Cette situation se présente chaque fois que l'on utilise un programme machine écrit pour le modèle 3 Qwerty et qui possède son propre pilote clavier. (c'est à dire qui ne fait pas appel à celui de la MEM). Exemples : SCRIPSIT, VISICALC, PROFILE, EDTASM, RSM, etc.

Plusieurs solutions s'offrent à nous :

1. Faire l'acquisition des versions "Azerty" des programmes qui ont déjà été convertis (à vrai dire, cette première solution ne s'offre pas à nous... mais s'achète !).

2. Modifier les programmes concernés en recherchant les pilotes claviers et en les remplaçant par des Azerty (facile à dire, mais difficile à faire...) ou en "déviant" ces pilotes claviers, et en les remplaçant par un appel à celui de la MEM (c'est ce qui a sans doute été fait par Tandy pour les programmes Scripsit et Visicalc... c'est très visible à l'utilisation du Visicalc Azerty qui se retrouve avec une fonction de répétition des touches et de déplacement du curseur désespérément lente...) Solution encore pas à la portée de tout le monde.

3. Reste une troisième solution (qui est la mienne et j'en ai un peu honte ...!!!) et qui consiste à repérer sur le clavier Azerty, l'emplacement des touches Qwerty ... et à coller de petites étiquettes. Le clavier numérique est alors "perdu" car remplacé par ! " = $ % & ' ( ) (**tableau 3**).

Les touches repérées "◤" n'ont aucune action (ne sont pas décodées par le pilote du clavier Qwerty).

Pilotes claviers : Les pilotes claviers Azerty et Qwerty possèdent la même adresse (3024H) mais sont différents... puisque des touches d'adresses diffé-

rentes dans la matrice clavier doivent retourner la même valeur ASCII (ce qui est encore heureux !). Un appui sur la touche Z par exemple, correspond :

en Qwerty à un 4 à l'adresse 3808H
en Azerty à un 128 à l'adresse 3804H

Dans les deux cas, un CALL 2BH doit retourner la valeur 5AH dans l'accumulateur.

ATTENTION : Toutes les touches des deux claviers donnent à la SORTIE DU CALL 49H les mêmes valeurs Hexa à l'exception de quelques unes...(tableau 4).

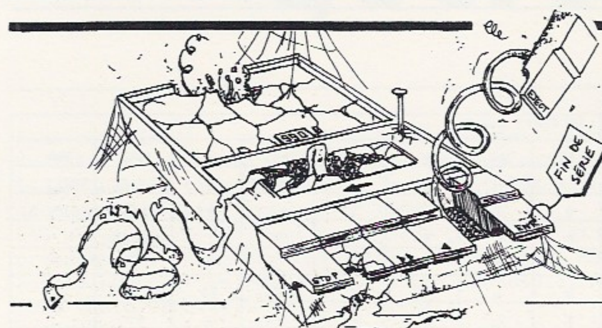
De plus, en Azerty, on obtient des valeurs supplémentaires correspondant aux minuscules accentuées, qui ne pouvaient pas être générées par le clavier Qwerty.

	QWERTY	AZERTY
Shift clear	1FH	aucune
Shift ©	60H	42H
Shift ENTER	0DH	60H

tableau 4

Les différences entre les codes générés par la frappe de : Shift clear, Shift enter, Shift + ©, sont dues au fait que la commutation majuscules/minuscules se fait par :
Shift Ø en Qwerty / Shift clear en Azerty.
L'arrêt provisoire d'un programme s'obtient par :
Shift © en Qwerty / Shift ENTER en Azerty.

Christian Triguéros



Mes expériences avec le CTR-80

L'utilisation d'un VU-METRE (*) ne suffit pas pour assurer une fiabilité totale à l'écriture-lecture sur cassettes.
Certaines cassettes, mal enroulées au départ, 'voilent' lors de la première utilisation. Après rebobinage, le voilage disparaît généralement, mais la lecture rectiligne d'un signal écrit en zig-zag donne des résultats désastreux.

Le remède est simple. Il suffit de faire un rebobinage total de la cassette avant la première utilisation.

Il faut également nettoyer périodiquement les têtes et les roulements à l'aide d'un produit du commerce ou simplement à l'alcool.
Mais il faut aussi savoir que la plupart des ennuis attribués à un manque de nettoyage proviennent généralement d'une simple poussière collée sur l'entrefer de la tête de lecture/écriture (la largeur de l'entrefer est inférieure à 5 microns !).

Ces poussières sont aisément éliminées en passant le petit doigt sur les têtes (une chiquenaude suffit). Cette manoeuvre est à faire absolument avant le chargement d'une nouvelle cassette. Les professionnels de la HI-FI ne procèdent pas autrement !

Certaines cassettes salissent littéralement les têtes en y agglomérant des dépôts. Il en résulte que les images magnétiques des 'Ø' et des '1' sur la bande sont alors "floues", comme sur une photo mal focalisée. Le signal lu est souvent très faible et déformé. Ces bandes doivent être impitoyablement rejetées, tout comme celles présentant des variations exagérées du niveau de lecture.

Chaque cassette devrait être préalablement inspectée, comme on le fait pour les disquettes pendant l'opération du formatage.

J'utilise à cet effet le court programme ci-après. Dans un premier temps, 70 chaînes de 240 'A' sont écrites sur la cassette en un peu moins de 10 minutes (à 500 bauds). Après rebobinage, le texte est relu et vérifié. Un STOP a lieu après chaque erreur, avec affichage du texte erroné.

Pour une sélection sévère, le niveau de lecture est dès le début ajusté aux 2/3 du réglage habituel du VU-METRE.

Lorsqu'une cassette passe le test, une autre cassette du même type, achetée le même jour dans le même rayon du même magasin le passe le plus souvent aussi.

L'inverse est aussi vrai. C'est mon seul critère de choix.

Un dernier conseil. Tracez au 'tipp ex' un point blanc bien visible sur la graduation 6 du potentiomètre de volume !

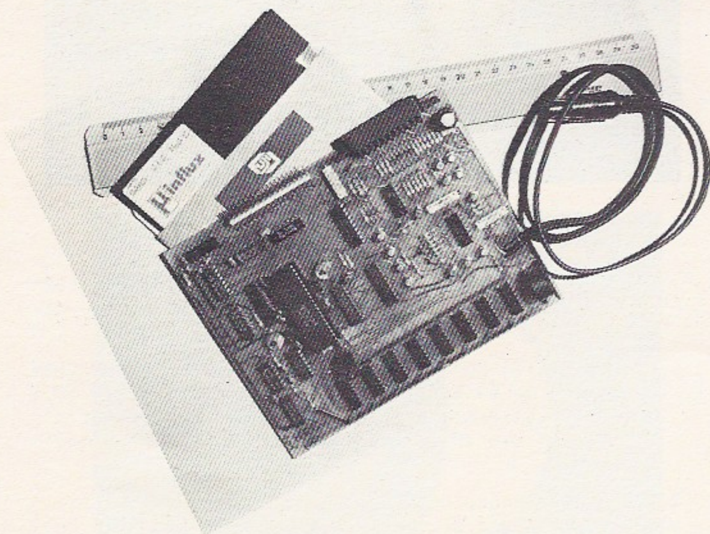
(*) décrit dans TRACE n° 5.




```
1 '-----------------------------------'  FF
2 '          Essai cassette           '  FE
3 '-----------------------------------'  FD
10 CLEAR 1000 : A$=STRING$(240,"A"):REMAD
20 CLS : PRINT "ESSAI K7." : INPUT "K7 -- RECORD -- ENTER ";Z$:R
EMEO
30 FOR I=1 TO 70 : PRINT @ 672,I : PRINT #-1,A$ :NEXT:REM4E
40 CLS : PRINT "REBOBINEZ. " : INPUT "K7 -- PLAY -- ENTER ";Z$:R
EMAC
50 FOR I=1 TO 70 : PRINT @ 672,I : INPUT #-1,B$:REM23
60 IF B$<>A$ PRINT @ 128,B$;"ERREUR..." : STOP:REMF5
70 NEXT:REM33
```

Roger Brousmiche

La carte HRC-80 nous en fait voir de toutes les couleurs...



Prix : 2 350 F ttc
Distributeur : Micro-Influx, 20 rue Laennec,
78330 Fontenay le Fleury

CONFIGURATION NECESSAIRE : Modèle 1 ou 3, VGS, 16 à 48 K, cassette ou disquette.

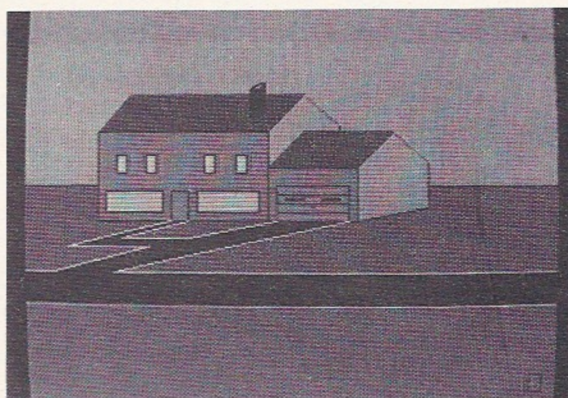
J'attendais avec une impatience non dissimulée (comme bien d'autres utilisateurs du TRS 80, je suppose) l'arrivée sur le marché d'une carte d'extension graphique/couleur donnant à mon TRS 80 modèle 1 une définition enfin digne de ses possibilités.
Ce jour tant attendu est enfin arrivé un beau matin de septembre où j'ai pris livraison de la carte HRC 80 commandée en juin 1983, et c'est avec une certaine fébrilité que j'ai procédé aux différents raccordements permettant de donner "Vie" au système. Ces connexions, au nombre de cinq, ne m'ont posé aucun problème :

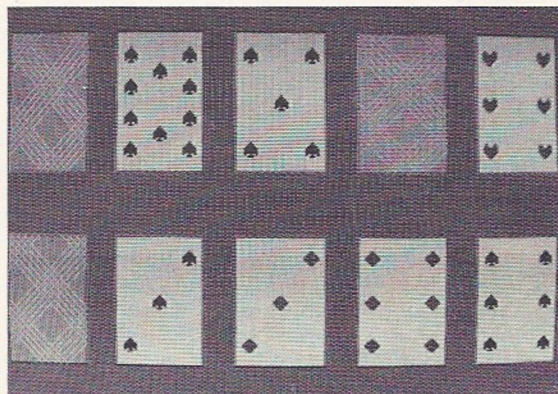
1 - Alimentation 5 V
2 - Sortie vidéo par prise DIN vers moniteur vidéo
3 - Entrée vidéo par cordon DIN en provenance du clavier du TRS 80
4 - Bus de données par nappe vers connecteur "bus" du TRS 80
5 - Cordon PERITEL pour liaison vers téléviseur couleur.

Il est à signaler que dans la version de base, la carte est livrée nue sans alimentation, avec la sortie VIDEO N et B et le câble de raccordement carte-clavier, mais sans le câble PERITEL et le bus de données ; ces différents accessoires peuvent être cependant obtenus chez le distributeur moyennant un léger supplément de prix.

Après avoir placé la disquette de démonstration dans le lecteur (une cassette est aussi disponible sur demande) et mis en route mon téléviseur couleur utilisé comme deuxième moniteur, j'ai pu admirer toutes les possibilités offertes par la carte HRC-80 :

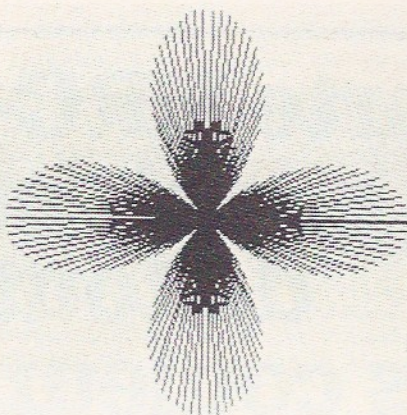
-Une résolution graphique de 320 points (horizontal) par 250 points (vertical), soit une capacité de 80000 points pour l'ensemble de l'écran : on est bien loin du 128 x 48 = 6144 points de la version de base et les diagonales tracées sont maintenant de vraies droites et non plus des "marches d'escalier" !!!

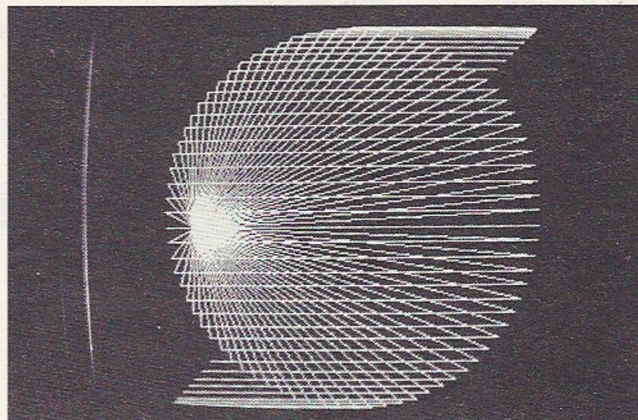




- Une disponibilité de 8 couleurs simultanées (noir, rouge, vert, jaune, bleu, magenta, cyan, blanc) à la fois pour le fond de couleur et le graphisme, soit 64 combinaisons (bien sûr, toutes ne sont pas forcément heureuses !!!). Ces 8 couleurs sont disponibles même avec la définition graphique maximale de 80000 points.

- La possibilité de mixer par instruction BASIC la basse résolution avec la haute résolution fournie par la carte HRC 80 et ceci par superposition d'images. Ceci présente l'avantage certain, par rapport aux systèmes avec mixage intégré graphique-texte (ATARI-800, COMMODORE VIC-20, CBM64, etc), de pouvoir effectuer un aiguillage sélectif de l'image par programme, par exemple : le graphisme haute résolution sur le moniteur couleur et le texte et le graphisme basse définition sur le moniteur N et B de base du TRS 80.





Les utilisateurs qui ne possèdent pas de moniteur couleur peuvent aussi bénéficier de la haute résolution sur l'écran noir et blanc de l'ordinateur. Les couleurs seront tout simplement remplacées par des teintes dégradées (entre le blanc et le noir). Le texte et la basse résolution habituels seront alors superposés avec l'image haute résolution, ce qui assure une entière compatibilité avec tous les logiciels existants.

- Le gros intérêt de la carte HRC-80 est qu'elle est fournie avec le logiciel correspondant de mise en oeuvre du graphisme haute résolution, à savoir un BASIC graphique dénommé GBAS/CMD qui offre à l'utilisateur, grâce à 10 instructions d'emploi aisé (à part peut-être HGET et HPUT) l'accès à une programmation facile d'applications graphiques (sans avoir recours au code machine encore réservé à une minorité). Ces instructions sont décrites succintement dans un manuel remis à l'utilisateur à la livraison de la carte, elles donnent accès :

- au choix de 4 modes d'écran (HDISP).
- au choix de 8 couleurs de fond d'écran (HCLS).
- au choix de 8 couleurs du graphisme (HCOLOR).
- au choix entre 2 pages graphiques (HPAGE).
- au tracé de points, droites, rectangles, rectangles colorés (HPLOT).
- au tracé de cercles (HCIRCLE) ou de huitièmes de cercles.
- au test de la couleur d'un point graphique (HPOINT).
- au transfert en mémoire graphique d'une image contenue dans un tableau entier (HPUT).
- au transfert dans un tableau entier d'une image en mémoire graphique (HGET).

Un petit regret cependant : ne pas disposer des instructions de type PAINT et DRAW qui font tout le charme de certains BASIC MICROSOFT (TRS-COLOR, DRAGON, SPECTRAVIDEO). Ceci n'est peut être que partie remise, puisque le code SOURCE GBAS/ASM est fourni avec la disquette et on peut ainsi espérer qu'un possesseur de la carte HRC-80 (en veine d'inspiration) "implémente" ces 2 instructions supplémentaires qui apparaîtront à beaucoup comme des nécessités à court terme.

Bien d'autres ressources inexploitées sont cachées dans la carte HRC-80 ; elles ne demandent qu'à être révélées par un utilisateur friand de code machine, maniant avec aisance le langage assembleur du Z80 et connaissant la programmation du processeur graphique TEXAS TMS 3536, véritable cerveau de la carte HRC-80.

Sans déflorer le sujet on peut citer les thèmes de réflexion suivants :

- présence de 4 générateurs de caractères programmables de 128 caractères chacun (caractères inscrits dans une matrice de 8 x 10).
- possibilité en utilisant ces générateurs de réaliser un affichage en mode VIDEOTEXT de 25 lignes x 40 caractères.
- disponibilité sur la carte HRC-80 d'une mémoire vive graphique de 64 K octets (2 pages graphiques de 30 K octets) utilisable peut-être en extension "mémoire virtuelle" des 48 Ko de la mémoire vive du système de base ?

Alors, possesseurs du HRC-80, à vos plumes, vos claviers, vos fers à souder ! La tâche est rude mais le résultat est à la mesure de vos efforts. Ne broyez plus du "NOIR", l'avenir est maintenant couleur "ARC EN CIEL" !

<u>En conclusion</u> nous ne pouvons que souhaiter "longue vie" (en rose) à la carte HRC-80 qui présente un excellent rapport qualité-prix et qui recèle encore autant de potentialités offrant la joie de la découverte à l'utilisateur.

<u>NOTA</u> : Certains utilisateurs d'anciens "modèles 1" devront procéder comme moi à une modification mineure à l'intérieur du clavier permettant à celui-ci de délivrer un signal VIDEO 50Hz et non plus 60Hz, faute de quoi la définition graphique de HRC-80 serait limitée à 320 x 210 = 67200 points au lieu de 80000 !!!

MODIFICATION 60Hz ⟶ 50Hz
----

1 - démontez votre clavier ;

2 - repérez Z32 (circuit 7493) : de la broche du circuit partent une piste vers la gauche et une piste vers la droite (encoche du circuit positionnée vers le haut).
Suivez la piste de droite et coupez-la juste avant le trou métallisé.

3 - soudez un fil faisant la liaison entre la broche 8 de Z32 et ce trou métallisé (broche 2 de Z66).

Votre TRS80 délivre maintenant un signal à fréquence trame égal à 50,08Hz utilisable par la carte HRC-80 avec une pleine résolution graphique.

<u>Dernière minute</u> : Nous apprenons que l'un des possesseurs de la carte HRC-80 vient de réaliser une fonction "PAINT" et en fait profiter les autres utilisateurs ! Aussi, la version actuelle de l'extension BASIC est délivrée maintenant avec cette nouvelle commande.

Remy Pineau

Phases de la lune



Si l'homme va sur la lune, c'est grâce aux ordinateurs... Avec le programme qui suit, nous ne vous proposons pas un voyage dans l'espace, mais tout simplement de visualiser sur l'écran de votre TRS le quartier de la lune tel que vous pourriez le voir dans le ciel, du moins si l'heure et les conditions météorologiques le permettent !

Aux environs de la nouvelle lune (phase proche de 29,5 ou voisine de 0) il est difficile de l'apercevoir car d'une part le croissant est petit et d'autre part la lune est proche du soleil et apparaît pendant la journée seulement.

Pendant la pleine lune (aux environs de la phase 15) la lune est visible toute la nuit et l'éclairage qu' elle procure est maximum.

Vers le premier quartier (phase 7 ou 8) elle brille le soir tandis qu'au dernier quartier (phase 21 à 22) elle est présente le matin.

– Le programme présenté calcule la phase de la lune et le jour de la semaine pour une date donnée (jour, mois, année). Il trace ensuite le croissant de lune correspondant sur écran.

– Il calcule d'abord le nombre de jours F depuis le début du calendrier grégorien suivant la méthode (ent () désigne la partie entière) :

Si mois ⩽ 2 alors mois = mois + 13 et année = année – 1 sinon mois = mois + 1.F = jour + ent (année ∗ 365, 25 + 0,1) + ent (mois ∗ 30,6 + 0,01) – ent (année/ 100) + ent (année/400) – 578164.
0,1 et 0,01 servent à éviter le décalage d'un jour provenant d'une erreur d'arrondi.

– La phase de la lune se calcule par rapport à une référence F0. Le jour F0, la nouvelle lune était à midi.

On calcule successivement :

AI = (F-F0)/29,53   nombre de lunaisons depuis F0
AH = 29,53 (AI – ent (AI))   phase de la lune

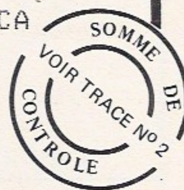
– De manière analogue, pour le jour de la semaine :
AI = (F-F0)/7 :   nombre de semaines depuis F0
AH = 7 (AI – ent (AI)) : jour de la semaine

– Le tracé de la lune est compris entre un cercle et une ellipse. Dans le programme, le cercle a un rayon de 22 x 2,1 suivant l'axe horizontal et de 22 suivant l'axe vertical.
Le coefficient 2,1 sert à tenir compte du rapport hauteur sur largeur des pavés graphiques sur l'écran.

Le rapport de dimension de l'axe horizontal de l'ellipse et du cercle est : cosinus (2πx phase/29,53). Suivant qu'on est dans la 1ère ou 2ème moitié de la lunaison, le croissant est tourné vers la droite ou la gauche (K = – 1 ou + 1).

```
1 ' Phase de la lune                                            ' FF
2 ' (C) Philippe Esclanson et TRACE                             ' FE
3 ' Configuration : modeles 1, 3, VGS. 16 a 48 K de RAM         ' FD
4 '------------------------------------------------------------ ' FC
10 CLS : DEFINT I-N : DIM Z$(6) : V=10 : A7=7:REMED
20 A=365.25 : VA=30.6 : AG=578164 : AJ=AG+2 : AT=29.530589 : F0=
144888:REMFE
30 PI=3.14159 : IN=22 : J=IN+3 : K0=63 : IR=IN*IN : CR=2.1:REM46

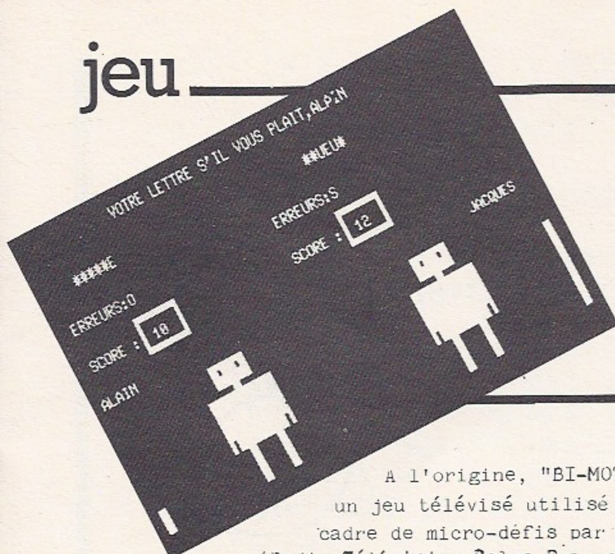
40 Z$(1)="LUNDI" : Z$(2)="MARDI" : Z$(3)="MERCREDI" : Z$(4)="JEU
DI":REM1B
50 Z$(5)="VENDREDI" : Z$(6)="SAMEDI" : Z$(0)="DIMANCHE":REMCA
60 '   C4
70 ' Debut du calendrier Gregorien : le 15 Octobre 1582  BA
80 ' correspond a F=0, d'ou valeur de AG
90 '   A6
100 ' Fin du calendrier Gregorien : le 4 Octobre 1582  9C
```



```
110 ' correspond a F=-1, d'ou valeur de AJ  92
120 '   88
130 PRINT @0,"DATE : (J,M,A)                   "; : PRINT @15,"";:REM9
B
140 INPUT IJ,IM,IA : PRINT @30,CHR$(31); : GOSUB 300:REMEU
150 AF=F-FO : AI=AF/AT : AH=AT*(AI-INT(AI)) : IH=INT(AH*V) : VH=
IH/V:REMCD
160 AI=AF/A7 : AH=A7*(AI-INT(AI)) : IH=INT(AH+0.1):REM49
170 PRINT @54,Z$(IH); : PRINT @64,"PHASE=";VH;:REM6D
180 '   4C
190 ' Trace du croissant de lune  42
200 '   38
210 AC=COS(2*PI*VH/AT) : K=-1 : IF VH>AT/2 THEN K=1:REMF3
220 FOR IY=-IN TO IN:REM73
230 IM=INT(CR*SQR(IR-IY*IY)) : KV=KO-AC*IM*K : KW=KO-IM*K : JY=I
Y+J:REM67
240 FOR JX=KW TO KV STEP K : SET(JX,JY) : NEXT JX:REM1F
250 NEXT IY : GOTO 130:REM82
260 '   FB
270 ' Calcul de F : nombre de jours ecoules  F1
280 ' depuis le debut du calendrier Gregorien  E7
290 '   DD
300 IF IM<=2 THEN IB=IA-1 : MA=IM+13 ELSE IB=IA : MA=IM+1:REMDO
310 VM=MA*VA+0.01 : I1=IB/100 : I4=I1/4:REM52
320 AP=IB*A+0.1 : F1=IJ+INT(AP)+INT(VM) : F=F1-AG-I1+I4:REM75
330 IF F<0 THEN F=F1-AJ ELSE RETURN:REMC4
340 IF F>=0 THEN PRINT @30,"CE JOUR N'A PAS EXISTE"; : F=0:REM7A

350 RETURN:REMOF
```

Philippe Esclangon



En deux mots...

A l'origine, "BI-MOTS" est un jeu télévisé utilisé dans le cadre de micro-défis par la RTBF (Radio Télévision Belge Francophone).

Le programme que je vous propose en est la réplique plus ou moins exacte. Le principe est basé sur le célèbre jeu du pendu mais se joue à deux, sur deux tableaux différents, ce qui lui ajoute un intérêt nouveau.

Après avoir demandé la longueur des mots à retrouver (de 6 à 9 lettres), l'ordinateur attend la proposition de lettre du premier joueur. Ce dernier doit répondre dans un temps donné matérialisé par un "baromètre" sur le côté de l'écran. Passé ce temps, le tour passe au second joueur. A chaque erreur, le "bonhomme" du joueur respectif descend d'un cran...

Le programme occupe environ 13 K de mémoire vive mais il peut être réduit en retirant les commentaires. Voici, à titre indicatif, l'utilisation des principales variables :

A$, B$: mots des deux joueurs.
BH$, BV$, BO$, BC$, BX$, BN$, BM$, EF$: utilisées pour les graphiques.
C$, D$: noms des deux joueurs.
E$, G$: dernière lettre entrée par les deux joueurs.
J$, R$: mot entré par un joueur mais qui s'adresse à l'autre.
P1, P2 : points des deux joueurs.
D : nombre de lettres des mots.
W, Z : compteurs pour la descente des personnages.

Tel qu'il est, le programme possède un vocabulaire de 80 mots (20 mots de 6, 7, 8 et 9 lettres).

A vous de jouer (et vite car l'ordinateur n'attend pas) !




```
10 '=============================================== F6
20 '---                  BI-MOTS                 --  EC
30 '---       Configuration : TRS-80 modele 1 et 3   --  E2
40 '---       (C) 1983 TRACE  ET Dubois Michael    --  D8
50 '=============================================== CE
60 '------------------ Affichage du titre ------------  C4
70 '  BA
80 CLEAR 4000 : DEFINT A-Z:REMF8
90 DIM EE$(24),GG$(24),M6$(20),M7$(20),M8$(20),M9$(20):REM04
100 FOR I=1 TO 20 : READ AA$ : M6$(I)=AA$ : NEXT:REM8B
110 FOR I=1 TO 20 : READ AA$ : M7$(I)=AA$ : NEXT:REM80
120 FOR I=1 TO 20 : READ AA$ : M8$(I)=AA$ : NEXT:REM75
130 FOR I=1 TO 20 : READ AA$ : M9$(I)=AA$ : NEXT:REM6A
140 CLS : PRINT CHR$(23):REM57
150 FOR I=256 TO 318:REMD5
160 PRINT @I,CHR$(191); : PRINT @I+256,CHR$(191); : NEXT:REMF1
170 PRINT @408,"BI-MOTS":REM5D
180 FOR I=1 TO 2000 : NEXT:REM3C
190 '---------------Affichage des regles du jeu----------  42
200 '  38
210 CLS:REMAA
220 PRINT @0,STRING$(63,CHR$(131)); : PRINT @15*64,STRING$(63,CH
R$(176));:REM52
230 SET(126,0) : SET(126,47) : FOR I=0 TO 47 : SET(0,I) : SET(12
7,I) : NEXT:REM51
240 PRINT @89,"B I  -  M O T S"; : PRINT @152,"PAR DUBOIS MICHAE
L";:REMFC
250 PRINT @346,"REGLES DU JEU"; : PRINT @410,"============="; :
PRINT @462,"PENDANT LE JEU IL NE FAUT EN AUCUN CAS";:REMD8
260 PRINT @532,"PRESSER LA TOUCHE <ENTER>";:REM53
270 PRINT @780,"PRESSER LA TOUCHE <CLEAR> POUR POURSUIVRE";:REMF
1
```

```
280 FOR I=1 TO 5000:REM95
290 IN$=INKEY$:REM84
300 IF IN$=CHR$(31) THEN 340:REMA7
310 NEXT I:REMF9
320 '-----------Introduction du nombre de lettres--------  BF
330 '  B5
340 CLS : PRINT @280,"AVANT-JEU";:REM2B
350 PRINT : PRINT @513,"ENTRER LE NOMBRE  DE LETTRES QUE VOUS DE
SIREZ  <6> <7> <8> <9> ";:REM9B
360 PRINT @593,; : INPUT D:REM96
370 IF D<6 OR D>9 THEN 350:REM28
380 A=D-(D-1):REMC7
390 ON D-5 GOTO 400,470,540,610:REM21
400 R1=RND(20):REM86
410 A$=M6$(R1) : IF A$="" THEN 400:REMOB
420 M6$(R1)="":REMC7
430 R2=RND(20):REM67
440 B$=M6$(R2) : IF B$="" THEN 430:REME7
450 M6$(R2)="":REMA8
460 GOTO 700:REMOF
470 R1=RND(20):REM40
480 A$=M7$(R1) : IF A$="" THEN 470:REMBD
490 M7$(R1)="":REM80
500 R2=RND(20):REM21
510 B$=M7$(R2) : IF B$="" THEN 500:REMA2
520 M7$(R2)="":REM60
530 GOTO 700:REMC8
540 R1=RND(20)
550 A$=M8$(R1) : IF A$="" THEN 540:REM77
560 M8$(R1)="":REM38
570 R2=RND(20):REMDA
580 B$=M8$(R2) : IF B$="" THEN 570:REM53
590 M8$(R2)="":REM19
600 GOTO 700:REM82
610 R1=RND(20):REMB3
620 A$=M9$(R1) : IF A$="" THEN 610:REM32
630 M9$(R1)="":REMF1
640 R2=RND(20):REM94
650 B$=M9$(R2) : IF B$="" THEN 640:REMOE
660 M9$(R2)="":REMD2
670 GOTO 700:REM3C
680 '-----------Introduction des prenoms des joueurs-----  56
690 '  4C
700 CLS:REMBE
710 PRINT : PRINT "ENTRER LE PRENOM DU 1ER JOUEUR             "
;C$ : PRINT : PRINT : PRINT : PRINT "ENTRER LE PRENOM DU 2EME JO
UEUR              ";D$:REM02
720 PRINT @95,""; : INPUT C$ : PRINT @352,""; : INPUT D$:REM5D
730 FOR I=1 TO 300 : NEXT:REM43
740 '---------------------Affichage du jeu----------------  1A
750 '  10
760 CLS:REM82
770 FOR A=1 TO D:REM32
780 PRINT @132+A,"*"; : PRINT @165+A,"*";:REMDB
790 NEXT A:REM1F
800 PRINT @258,"ERREURS:" : PRINT @287,"ERREURS:":REMDB
810 PRINT @386,"SCORE :" : PRINT @415,"SCORE :":REM2E
820 PRINT @513,C$; : PRINT @566,D$;:REMD4
830 FOR X=20 TO 30 : SET(X,16) : NEXT X : FOR X=77 TO 88 : SET(X
,16) : NEXT X : FOR Y=16 TO 21 : SET(77,Y) : NEXT Y : FOR Y=16 T
O 21 : SET(19,Y) : NEXT Y:REMD6
```



```
840 FOR X=20 TO 30 : SET(X,21) : NEXT X : FOR X=77 TO 88 : SET(X
,21) : NEXT X : FOR Y=16 TO 21 : SET(30,Y) : NEXT Y : FOR Y=16 T
O 21 : SET(88,Y) : NEXT Y:REMD9
850 P1=10:REMF4
860 PRINT @394,P1;:REM27
870 P2=10:REMDF
880 PRINT @423,P2;:REM19
890 '---------------Construction des personnages----------  83
900 '  79
910 EF$=STRING$(64,CHR$(128)):REM5D
920 BH$=CHR$(188)+CHR$(172)+CHR$(188)+CHR$(156)+CHR$(188):REM2D
930 BV$=CHR$(143)+CHR$(143)+CHR$(191)+CHR$(143)+CHR$(143):REM33
940 BO$=STRING$(9,CHR$(128)):REM6A
950 BC$=STRING$(9,CHR$(191)):REM6C
960 BX$=CHR$(151)+CHR$(191)+CHR$(191)+CHR$(191)+CHR$(191)+CHR$(1
91)+CHR$(191)+CHR$(191)+CHR$(171):REM4D
970 BN$=CHR$(191)+CHR$(187)+CHR$(191)+CHR$(183)+CHR$(191):REM01
980 BM$=CHR$(176)+CHR$(176)+CHR$(179)+CHR$(179)+CHR$(191)+CHR$(1
79)+CHR$(179)+CHR$(176)+CHR$(176):REM1A
990 PRINT @847,CHR$(191); : PRINT @851,CHR$(191); : PRINT @911,C
HR$(191); : PRINT @915,CHR$(191); : PRINT @975,CHR$(191); : PRIN
T @979,CHR$(191);:REMCC
1000 PRINT @527,BH$; : PRINT @591,BV$; : PRINT @653,BC$; : PRINT
 @717,BC$; : PRINT @781,BX$;:REMDC
1010 '--------------------------------------------------------  0B
1020 PRINT @556,BH$; : PRINT @620,BV$; : PRINT @682,BC$; : PRINT
@746,BC$; : PRINT @746,BC$; : PRINT @810,BX$;:REMF3
1030 PRINT @876,CHR$(191); : PRINT @880,CHR$(191); : PRINT @880,
CHR$(191); : PRINT @940,CHR$(191); : PRINT @944,CHR$(191); : PRI
NT @1004,CHR$(191); : PRINT @1008,CHR$(191);:REM45
1040 '-------------------Initialisation-------------------  EC
1050 '  E2
1060 A=0:AT=0:C=0:CH=0:CP=0:E=0:F=0:G=0:H=0:I=0:J=0:K=0:L=0:N=0:
O=0:PO=0:Q=0:R=0:R1=0:R2=0:S=0:T=0:TH=0:TI=0:TX=0:TZ=0:V=0:W=0:Y
=0:Z=0:REMB5
1070 '----------Introduction d'une lettre du 1er joueur---  CE
1080 '  C4
1090 GOSUB 2030 : GOSUB 2090:REMCE
1100 IF GH>=D THEN TH=1:REM39
1110 IF GH>=D THEN 2480 ELSE PRINT @0,EF$; : PRINT @13,"VOTRE LE
TTRE S'IL VOUS PLAIT,";C$:REMF8
1120 FOR I=1 TO 200 : NEXT:REMBC
1130 GOSUB 1230:REM3B
1140 '--------Introduction d'une lettre du 2eme joueur----  88
1150 '  7E
1160 IF CP>=D THEN TI=1:REMF8
1170 IF CP>=D THEN 2630 ELSE PRINT @0,EF$; : PRINT @13,"VOTRE LE
TTRE S'IL VOUS PLAIT,";D$:REMBA
1180 FOR I=1 TO 200 : NEXT:REM80
1190 GOSUB 1340:REMFD
1200 GOSUB 2090 : GOTO 1160:REM61
1210 '------------Decompte du temps du 1er joueur---------  42
1220 '  38
1230 E$="":REMAC
1240 FOR Y=33 TO 47:REME7
1250 RESET(2,Y) : RESET(3,Y):REMCB
1260 FOR I=1 TO 170 : NEXT I:REME1
1270 E$=INKEY$:REMFF
1280 IF E$=""THEN 1290 ELSE GOTO 2150:REM30
1290 NEXT Y:REM11
1300 IF E$="" THEN E$="." : GOTO 1460:REMCA
```



```
1310 END
1320 '-----------Decompte du temps du 2eme Joueur---------  D3
1330 '   C9
1340 G$="":REM3B
1350 FOR Y=33 TO 47:REM78
1360 RESET(124,Y) : RESET(125,Y):REM92
1370 FOR I=1 TO 170 : NEXT I:REM72
1380 G$=INKEY$:REM8E
1390 IF G$="" THEN 1400 ELSE 2320:REM55
1400 NEXT Y:REMA3
1410 IF G$="" THEN G$="." : GOTO 1760:REM55
1420 '----------------Descente du bonhomme (1)------------  6F
1430 '   65
1440 PRINT @0,EF$; : PRINT @16,E$; : PRINT @18,"NE SE TROUVE PAS
 DANS LE MOT";:REME5
1450 FOR I=1 TO 200 : NEXT:REM71
1460 P1=P1-1:REM71
1470 Z=Z+1:REMB6
1480 IF P1<10 THEN PRINT @395,P1; ELSE PRINT @394,P1;:REM5E
1490 IF Z=1 THEN PRINT @266,E$; ELSE 1510:REM7A
1500 PRINT @527,BO$; : PRINT @591,BH$; : PRINT @653,BO$; : PRINT
 @655,BV$; : PRINT @717,BC$; : PRINT @781,BC$; : PRINT @845,BX$;
:REM15
1510 IF Z=2 THEN PRINT @267,E$; ELSE 1530:REM62
1520 PRINT @590,BO$; : PRINT @717,BO$; : PRINT @655,BH$; : PRINT
 @719,BV$; : PRINT @781,BC$; : PRINT @845,BC$; : PRINT @909,BX$;
:REMFB
1530 IF Z=3 THEN PRINT @268,E$; ELSE 1550:REM4A
1540 PRINT @655,BO$; : PRINT @781,BO$; : PRINT @719,BH$; : PRINT
 @783,BV$; : PRINT @845,BC$; : PRINT @909,BC$; : PRINT @973,BX$;
:REMDE
1550 IF Z=4 THEN PRINT @269,E$; ELSE 1580:REM30
1560 PRINT @719,BO$;:REM33
1570 PRINT @783,BN$; : PRINT @845,BM$; : PRINT @909,BC$; : PRINT
@973,BC$;:REM85
1580 IF Z=5 THEN PRINT @270,E$; ELSE 1600:REM20
1590 PRINT @783,BO$; : PRINT @845,BO$; : PRINT @783,BH$; : PRINT
 @847,BV$; : PRINT @973,BC$;:REM77
1600 IF Z=6 THEN PRINT @271,E$; ELSE 1620:REM08
1610 PRINT @783,BO$; : PRINT @845,BO$; : PRINT @909,BO$; : PRINT
 @847,BN$; : PRINT @909,BM$;:REM5B
1620 IF Z=7 THEN PRINT @272,E$; ELSE 1640:REMF0
1630 PRINT @845,BO$; : PRINT @909,BO$; : PRINT @911,BN$; : PRINT
 @973,BM$;:REM38
1640 IF Z=8 THEN PRINT @273,E$; ELSE 1660:REMD8
1650 PRINT @909,BO$; : PRINT @973,BO$; : PRINT @911,BH$; : PRINT
 @975,BV$;:REM1D
1660 IF Z=9 THEN PRINT @274,E$; ELSE 1690:REMBF
1670 PRINT @909,BO$; : PRINT @975,BN$;:REMD8
1680 IF W>=10 THEN TZ=1:REMFC
1690 IF Z>=10 THEN PRINT @275,E$; : PRINT @975,BO$; : IF TZ=1 TH
EN 2780 ELSE GOTO 2780:REM11
1700 GOSUB 2030:REM00
1710 GOTO 1160:REMF7
1720 '-------------Descente du bonhomme (2)--------------  42
1730 '   38
1740 PRINT @0,EF$; : PRINT @16,G$; : PRINT @18,"NE SE TROUVE PAS
 DANS LE MOT";:REMB6
1750 FOR I=1 TO 200 : NEXT:REM44
1760 P2=P2-1:REM42
1770 W=W+1:REM8F
```



```
1780 IF P2<10 THEN PRINT @424,P2; ELSE PRINT @423,P2;:REM3C
1790 IF W=1 THEN PRINT @295,G$; ELSE 1810:REM49
1800 PRINT @556,BO$; : PRINT @682,BO$; : PRINT @620,BH$; : PRINT
 @684,BV$; : PRINT @810,BC$; : PRINT @874,BX$;:REMC8
1810 IF W=2 THEN PRINT @296,G$; ELSE 1830
1820 PRINT @620,BO$; : PRINT @746,BO$; : PRINT @684,BH$; : PRINT
 @748,BV$; : PRINT @874,BC$; : PRINT @938,BX$;:REMA5
1830 IF W=3 THEN PRINT @297,G$; ELSE 1850:REM18
1840 PRINT @682,BO$; : PRINT @746,BO$; : PRINT @810,BO$; : PRINT
 @748,BH$; : PRINT @812,BV$; : PRINT @938,BC$; : PRINT @1002,BX$
;:REM8F
1850 IF W=4 THEN PRINT @298,G$; ELSE 1870:REM0D
1860 PRINT @746,BO$; : PRINT @874,BO$; : PRINT @812,BN$; : PRINT
 @874,BM$; : PRINT @1002,BC$;:REM51
1870 IF W=5 THEN PRINT @299,G$; ELSE 1890:REME8
1880 PRINT @810,BO$; : PRINT @874,BO$; : PRINT @812,BH$; : PRINT
 @876,BV$;:REM3F
1890 IF W=6 THEN PRINT @300,G$; ELSE 1910:REMEB
1900 PRINT @810,BO$; : PRINT @938,BO$; : PRINT @876,BN$; : PRINT
 @938,BM$;:REM24
1910 IF W=7 THEN PRINT @301,G$; ELSE 1930:REMD3
1920 PRINT @874,BO$; : PRINT @938,BO$; : PRINT @1002,BO$; : PRIN
T @940,BN$; : PRINT @1002,BM$;:REME4
1930 IF W=8 THEN PRINT @302,G$; ELSE 1950:REMBB
1940 PRINT @1002,BO$; : PRINT @940,BH$; : PRINT @1004,BV$;:REMA2

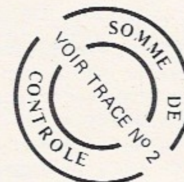
1950 IF W=9 THEN PRINT @303,G$; ELSE 1980:REMA2
1960 PRINT @938,BO$; : PRINT @1004,BN$;:REM93
1970 IF Z>=10 THEN TX=1:REMD8
1980 IF W>=10 THEN PRINT @304,G$; : PRINT @1002,BO$; : IF TX=1 T
HEN 2780 ELSE GOTO 2780:REMDA
1990 GOSUB 2090:REMD7
2000 GOTO 1100:REMDA
2010 '---------Construction du 1er compte a rebours-------  1F
2020 '  15
2030 FOR X=2 TO 3 : FOR Y=33 TO 47:REMC4
2040 SET(X,Y) : NEXT Y : NEXT X:REM1D
2050 RETURN:REM64
2060 '-------------------------------------------------  EC
2070 '---------Construction du 2eme compte a rebours------  E2
2080 '  D8
2090 FOR X=124 TO 125 : FOR Y=33 TO 47:REMBD
2100 SET(X,Y) : NEXT Y : NEXT X:REME0
2110 RETURN:REM28
2120 '-------------------------------------------------  B0
2130 'Comparation des lettres entrees et verification (1)-  A6
2140 '  9C
2150 F=F+1 : IF F=1 THEN 2190:REM88
2160 FOR H=1 TO R:REMAA
2170 IF E$=EE$(H) THEN PRINT @0,EF$; : PRINT @10,"VOUS AVEZ DEJA
 ENONCE CETTE LETTRE !"; : FOR I=1 TO 200 : NEXT I : GOSUB 2030
: GOTO 1100:REMFF
2180 NEXT H:REMA5
2190 R=R+1:REMF3
2200 EE$(R)=E$:REMD1
2210 FOR T=1 TO D:REM7A
2220 F$=MID$(A$,T,1):REM80
2230 IF E$=F$ THEN PRINT @0,EF$; : PRINT @19,E$; : PRINT @21,"ES
T UN BONNE LETTRE"; : FOR I=1 TO 200 : NEXT I : PRINT @133+(T-1)
,E$; : GH=GH+1 : P1=P1+1 : IF P1<10 THEN PRINT @395,P1; ELSE PRI
NT @394,P1; : A=A+(D/D):REME1
```

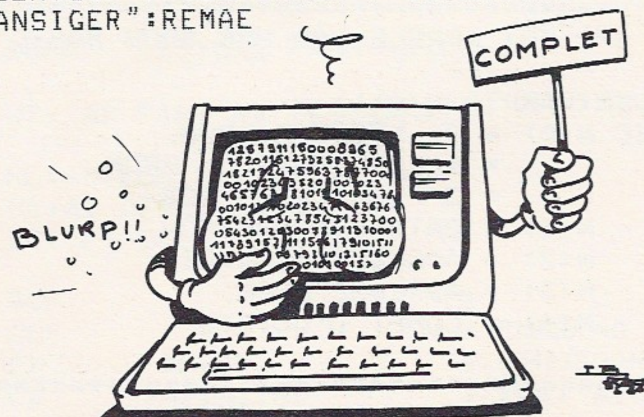


```
2240 NEXT T:REM5D
2250 FOR PO=1 TO LEN(A$)-1+1:REMA5
2260 IF E$=MID$(A$,PO,1) THEN 2280:REME9
2270 NEXT PO : PO=0:REM16
2280 IF PO=0 THEN 1440 ELSE GOSUB 2030 : GOTO 1100:REM9C
2290 '-----------------------------------------------------  06
2300 'Comparaison des lettres entrees et verification (2)-  FC
2310 '   F1
2320 C=C+1 : IF C=1 THEN 2360:REME7
2330 FOR G=1TOK:REM07
2340 IF G$=GG$(G) THEN PRINT @0,EF$; : PRINT @10,"VOUS AVEZ DEJA
 ENONCE CETTE LETTRE !"; : FOR I=1 TO 300 : NEXT : GOSUB 2090 :
GOTO 1160:REM8B
2350 NEXT G:REMFB
2360 K=K+1:REM56
2370 GG$(K)=G$:REM27
2380 FOR CH=1 TO D:REM98
2390 T$=MID$(B$,CH,1):REM8F
2400 IF G$=T$ THEN PRINT @0,EF$; : PRINT @19,G$; : PRINT @21,"ES
T UNE BONNE LETTRE"; : FOR I=1 TO 200 : NEXT I : PRINT @166+(CH-
1),G$; : CP=CP+1 : P2=P2+1 : IF P2<10 THEN PRINT @424,P2; ELSE P
RINT @423,P2; : AT=AT+(D/D)
2410 NEXT CH:REM7B
2420 FOR O=1 TO LEN (B$):REM46
2430 IF G$=MID$(B$,O,1) THEN 2450:REM8C
2440 NEXT O : O=0:REM0B
2450 IF O=0 THEN 1740 ELSE GOSUB 2090 : GOTO 1160:REM32
2460 '--------Introduction du mot de l'adversaire (1)-----  5B
2470 '   51
2480 IF TI=1 THEN 2780 ELSE PRINT @0,EF$; : PRINT @18,"REPONDEZ-
VOUS (O OU N) ?";:REM81
2490 FOR Y=33 TO 47 : RESET(2,Y) : RESET(3,Y) : FOR I=1 TO 170 :
 NEXT I:REM0E
2500 L$=INKEY$:REM25
2510 IF L$="" THEN 2520 ELSE 2540:REME4
2520 NEXT Y:REM3F
2530 IF L$="" THEN PRINT @0,EF$; : PRINT @15,"VOTRE TEMPS EST EC
OULE !"; : E$="." : GOTO 1450:REM60
2540 PRINT @43,L$; : IF L$="O" THEN 2550 ELSE GOSUB 2030 : GOTO
1160:REMF0
2550 FOR I=1 TO 200 : NEXT I : PRINT @0,EF$; : PRINT @15,"VOTRE
REPONSE "; : INPUT J$:REM4C
2560 IF J$=B$ THEN P1=P1+5 ELSE GOTO 2580:REMF0
2570 IF P1<10 THEN PRINT @395,P1; ELSE PRINT @394,P1;:REM17
2580 IF J$<>B$ THEN P1=P1-5:REM31
2590 IF P1<10 THEN PRINT @395,P1; ELSE PRINT @394,P1;:REM03
2600 GOTO 2780:REM70
2610 '--------Introduction du mot de l'adversaire (2)-----  C4
2620 '   BA
2630 IF TH=1 THEN 2780 ELSE PRINT @0,EF$; : PRINT @18,"REPONDEZ-
VOUS (O OU N) ?";:REMEB
2640 FOR Y=33 TO 47 : RESET(124,Y) : RESET(125,Y) : FOR I=1 TO 1
70 : NEXT I:REMAD
2650 M$=INKEY$:REM8D
2660 IF M$="" THEN 2670 ELSE 2690:REM40
2670 NEXT Y:REMA8
2680 IF M$="" THEN PRINT @0,EF$; : PRINT @15,"VOTRE TEMPS EST EC
OULE !"; : G$="." : GOTO 1750:REMC3
2690 PRINT @43,M$; : IF M$="O" THEN 2700 ELSE GOSUB 2090 : GOTO
1100:REM5A
2700 FOR I=1 TO 200 : NEXT I : PRINT @0,EF$; : PRINT @15,"VOTRE
```

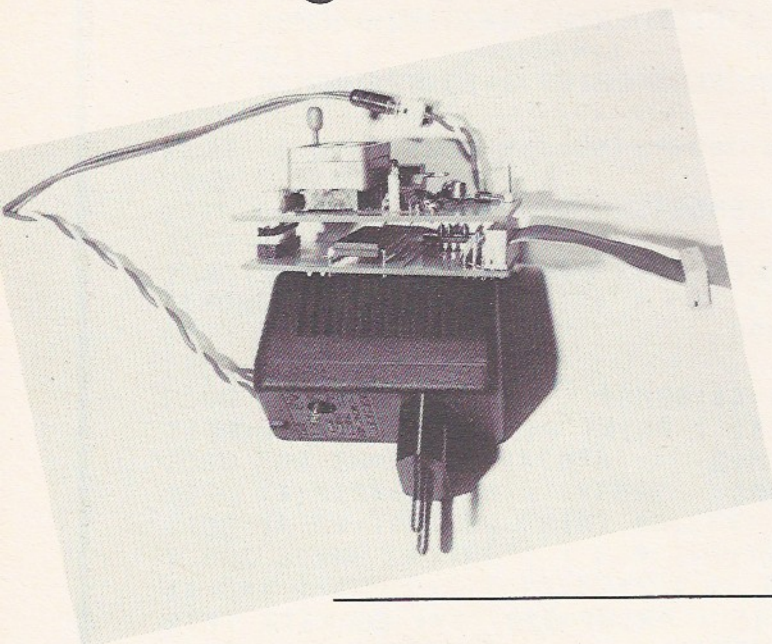


```
REPONSE "; : INPUT R$:REMAD
2710 IF R$=A$ THEN P2=P2+5 ELSE GOTO 2730:REM54
2720 IF P2<10 THEN PRINT @424,P2; ELSE PRINT @423,P2;:REM8C
2730 IF R$<>A$ THEN P2=P2-5:REM92
2740 IF P2<10 THEN PRINT @424,P2; ELSE PRINT @423,P2;:REM78
2750 GOTO 2780:REMDA
2760 '--------------------Fin du jeu----------------------  2E
2770 '   24
2780 IF Z=10 AND W=10 THEN 2800:REM08
2790 IF P1=P2 THEN 2890:REM0C
2800 PRINT @0,EF$; : PRINT @17,"LES MOTS ETAIENT :"; : PRINT @69
,A$; : PRINT @102,B$;:REM31
2810 FOR I=1 TO 500 : NEXT I:REMD0
2820 IF P1=P2 THEN 2890:REMED
2830 IF P1<P2 THEN N$=D$ ELSE N$=C$:REM89
2840 FOR N=1 TO 10 : PRINT @0,EF$; : PRINT @10,"*** C'EST ";N$;"
 QUI REMPORTE LA PARTIE ***";:REM13
2850 FOR I=1 TO 400 : NEXT I:REMA8
2860 PRINT @0,EF$;:REM91
2870 FOR C=1 TO 100 : NEXT C:REMA3
2880 NEXT N : GOTO 2920:REM4C
2890 PRINT @0,EF$; : FOR L=1 TO 15 : PRINT @20,"*** EGALITE ?!?
***"; : FOR I=1 TO 300 : NEXT I:REM10
2900 PRINT @0,EF$;:REM69
2910 FOR I=1 TO 200 : NEXT I : NEXT L:REM61
2920 PRINT @0,EF$;:REM55
2930 PRINT @15,"UNE AUTRE PARTIE ?";:REM13
2940 PRINT @32,; : INPUT O$:REM85
2950 FO$=LEFT$(O$,1):REMC8
2960 IF FO$="O" THEN 210 ELSE 2970:REMB7
2970 CLS : END:REM1D
2980 DATA "GUERRE","CHEVAL","KIMONO","TOMATE","VALISE","BATEAU",
"BANQUE","SOLDAT","HOMMES","FLEUVE","JOUEUR","DISQUE","MIROIR","
NUMERO","COURSE","TENNIS","HOCKEY","AVENUE","LATTES","NAGEUR":RE
M93
2990 DATA "POMPIER","CONSEIL","MEMOIRE","ARTISTE","BOUQUET","AUT
OBUS","RIVIERE","OUVREUR","SPECTRE","MUSIQUE","EDITION","TAUREAU
","SCIENCE","PLANETE","JOURNAL","MAGASIN","PLONGER","CIRCUIT","R
ANCART","FENETRE":REM84
3000 DATA "NATATION","JONCTION","CENDRIER","AVIATION","LUMINEUX"
,"COMPTEUR","POSITION","ARGUMENT","VARIABLE","FONCTION","DISCORD
E","RELATION","LONGUEUR","RESULTAT","ONDOYANT","PRATIQUE","FAUTE
UIL","ECRIVAIN","COMMANDE","CASSETTE":REM9D
3010 DATA "EMPREINTE","BOULANGER","BAIGNOIRE","TELEPHONE","CIGAR
ETTE","FORMATION","IMPULSION","PROGRAMME","PRECISION","OPERATEUR
","EXECUTION","EXTENSION","RESSOURSE","GRAPHIQUE","CONSTANTE","R
ABATTEUR","INGENIEUR","CONFORMER","OUVERTURE":REM9A
3020 DATA "TRANSIGER":REMAE
```





Michaël Dubois



Ouvrez votre modèle 1 sur le monde extérieur (IV)

Dans cette série d'articles, nous avons tout d'abord réalisé une interface parallèle 24 bits pour le modèle 1, puis nous avons choisi de prendre comme exemple d'application, un "brûleur" de mémoires EPROM qui pourra toujours rendre service aux bricoleurs. Nous vous présentons aujourd'hui le dernier volet de cette série qui concerne cette fois le logiciel nécessaire pour piloter cette application. La photo montre la carte "brûleur d'EPROM" enfichée sur la carte d'interface parallèle. L'alimentation est constituée par un chargeur secteur pour calculette, délivrant une tension continue de 5 volts.

Le programme se compose de deux parties : le corps principal en Basic, et une routine en assembleur pour la gestion des temps d'écriture des signaux électriques vers la mémoire EPROM. Signalons que ces temps doivent être très précis et sont imposés par les constructeurs d'EPROM. Ils sont donnés ici pour une mémoire de type 2716 ou équivalent et les temporisations tiennent compte de la fréquence d'horloge du TRS modèle 1 (1,77 Mhz).

Ce logiciel est prévu pour une configuration disquette avec 48 Ko de MEV. Pour d'autres configurations il sera donc nécessaire de retoucher l'adresse de début d'implantation de la routine en langage machine (8000 hexa dans notre cas) et de remplacer DEFUSR par 2 POKE dans le cas du Basic cassette. La routine en langage machine reste intacte et peut être relogée n'importe où en mémoire. Elle devra naturellement être chargée avant le programme Basic (LOAD sous DOS, ou SYSTEM en version cassette). Pensez à prévoir un "memory size" en conséquence.

La zone tampon à écrire dans la 2716 peut se trouver à une adresse quelconque en mémoire. Elle peut être chargée par différents moyens dont avez le choix : assembleur, utilitaire langage machine, POKE en Basic, etc. Il est aussi possible de compléter le programme ci-dessous pour remplir automatiquement cette zone à partir d'un fichier cassette ou disquette. Nous vous laissons faire...! Dernier détail : les adresses demandées par le programme sont exprimées en héxadécimal.

```
00010 ;**************************************************
00020 ;*                                                *
00030 ;*        LECTURE COMPARE ECRITURE PROM 2716      *
00040 ;*                                                *
00050 ;*================================================*
00060 ;*  CE S/P EST APPELE PAR USR SOUS BASIC II       *
00070 ;*                                                *
00080 ;*  X=USR(VARPTR(N(0)))                           *
00090 ;*  AVEC N(0) = 0  LECTURE PROM                   *
00100 ;*            = 1  COMPARAISON PROM               *
00110 ;*            = 2  ECRITURE PROM                  *
00120 ;*       N(1) = DATA COMPARAISON                  *
00130 ;*       N(2) = ADRESSE MEMOIRE                   *
00140 ;*       N(3) = ADRESSE PROM                      *
00150 ;*       N(4) = COMPT D'OCTETS                    *
00160 ;*                                                *
00170 ;**************************************************
```

```
00180 ;
00190 ;        VERSION : 26/11/83
00200 ;
00210 ADROUT   EQU     08000H          ; ADRESSE ROUTINE
00220 PORTA    EQU     10H             ; ADR 8255 PORT A
00230 PORTB    EQU     11H             ; ADR 8255 PORT B
00240 PORTC    EQU     12H             ; ADR 8255 PORT C
00250 PCNTL    EQU     13H             ; ADR 8255 CONTROLE
00260 PMODE    EQU     92H             ; MODE 8255 INIT
00270 INITRD   EQU     82H             ; MODE 8255 LECTURE PROM
00280 INITWR   EQU     80H             ; MODE 8255 ECRITURE PROM
00290 PGM      EQU     02H             ;
00300 NPGM     EQU     03H             ; BIT 1 PORT C=1 SELECT PROM
00310 TIMWP    EQU     0CE0H           ; TEMPO 50 MS ECR PROM
00320 TEMPO    EQU     060H            ; TEMP BAS IN BC=LG TEMP
00330 ;
00340          ORG     ADROUT
00350 ;
00360 DEBUT    CALL    0A7FH           ; APPELE ADRESSE
00370          LD      IX,0000         ;
00380          ADD     IX,SP           ; SAUVEGARDE ADR PILE -> IX
00390          LD      SP,HL           ; ADR TABLEAU -> SP
00400          EXX                     ; SAUVEGARDE ADR TABLEAU
00410          POP     BC              ;
00420          LD      A,C             ; N(0) -> A
00430          POP     HL              ;
00440          POP     HL              ; N(2) -> HL
00450          POP     DE              ; N(3) -> DE
00460          POP     BC              ; N(4) -> BC
00470          LD      SP,IX           ; REMISE A JOUR DE SP
00480          OR      A               ;
00490          JR      Z,LECPR         ; SI N(0)=0
00500          CP      2               ;
00510          JR      Z,ECRPR         ; SI N(0)=2
00520 ;
00530 ;***** COMPR - COMPARE PROM  N(0)=1
00540 ;
00550 ; AD DEBUT,AD FIN, AD MEMOIRE
00560 ;
00570 ; IN     HL:AD COMP MEM    N(2)
00580 ;        DE:AD DEBUT PROM N(3)
00590 ;        BC:CPT OCTET      N(4)
00600 ;
00610 COMPR    SCF                     ; CARRY=1
00620          JR      LEC10           ;
00630 ;
00640 ;***** LECPR - LECTURE PROM  N(0)=0
00650 ;
00660 ; AD DEBUT, AD FIN, AD DESTINATION
00670 ;
00680 ; IN     HL:AD ECR MEM     N(2)
00690 ;        DE:AD DEBUT PROM N(3)
00700 ;        BC:CPT OCTET      N(4)
00710 ;
00720 LECPR    SCF                     ;
00730          CCF                     ; CARRY=0
00740 LEC10    EX      AF,AF'          ; SAUVEGARDE DE CARRY
00750 LEC20    LD      A,INITRD        ; PORT A,C=OUT B=IN
```

```
                OUT     (PCNTL),A       ; INIT 8255
        LEC30   PUSH    DE              ; SAUVEGARDE ADR PROM
                LD      A,E             ;
                OUT     (PORTA),A       ; ECR POIDS FAIBLE ADR
                LD      (3C3CH),A       ; AFFICHAGE ECRAN
                SLA     D               ;
                SLA     D               ;
                SLA     D               ; CADRAGE POIDS FORT ADR
                IN      A,(PORTC)       ; LECTURE PORT C
                AND     07H             ; PROTECTION BITS PORT C
                OR      D               ; BIT 0-2 + ADR P FORT -> A
                OUT     (PORTC),A       ; ECR ADR P FORT PROM
                POP     DE              ; RESTITUTION ADR PROM N(3)
                LD      A,NPGM          ;
                OUT     (PCNTL),A       ; SELECTION PROM
                IN      A,(PORTB)       ; LECTURE PROM
                PUSH    AF              ; SAUVEGARDE DES DATA
                LD      A,PGM           ;
                OUT     (PCNTL),A       ; DESSELECTION PROM
                POP     AF              ; RESTITUTION DATA  -> A
                EX      AF,AF'          ; RESTITUTION DE CARRY
                JR      C,LEC40         ; SI CARRY=1 -> COMPARAISON
                EX      AF,AF'          ; RESTITUTION DES DATAS
                LD      (HL),A          ; ECR DATA A L'ADR MEM N(2)
                JR      LEC60           ; SUITE
        LEC40   EX      AF,AF'          ; RESTITUTION DES DATA
                CP      (HL)            ; COMPAR DATA AVEC (N(2))
                JR      Z,LEC60         ; SAUT SI PAS D'ERREUR
        LEC50   EXX                     ; RESTITUTION ADR TABLEAU
                LD      BC,000A         ; REAJUSTEMENT DU SP
                ADD     HL,BC           ; ADR DE N(4)
                LD      SP,HL           ; ADR TABLEAU -> SP
                EXX                     ; REG A SAUVER
                PUSH    BC              ; BC -> N(4)
                PUSH    DE              ; DE -> N(3)
                PUSH    HL              ; HL -> N(2)
                LD      D,(HL)          ; DATA MEM -> D
                LD      E,A             ; DATA PROM -> E
                PUSH    DE              ; DE -> N(1)
                LD      SP,IX           ; RESTITUTION DU SP
                RET                     ; RETOUR AU BASIC
        LEC60   INC     HL              ; INCREMENT DE  N(2)
                INC     DE              ; INCREMENT DE N(3)
                DEC     BC              ; DECREMENT DE N(4)
                LD      A,B             ;
                OR      C               ; TEST SI N(4)=0
                JR      NZ,LEC30        ; SAUT SI <> 0
                EX      AF,AF'          ; RESTITUTION DE CARRY
                JR      C,LEC50         ; FIN DU MODULE COMPARE
                RET                     ; RETOUR AU BASIC
        ;
        ;***** ECRPR - ECRITURE PROM  N(0)=2
        ;
        ; AD DEBUT, AD FIN, AD PROM
        ;
        ; IN    HL:AD DEBUT MEM N(2)
        ;       DE:AD ECR PROM  N(3)
```

```
01330 ;          BC:CPT OCTET    N(4)
01340 ;
01350 ECRPR   LD      A,INITWR          ; PORT A,B,C=OUT
01360         OUT     (PCNTL),A         ; INIT 8255 ECRITURE
01370         LD      A,NPGM            ; PGM=0
01380         OUT     (PCNTL),A         ;
01390         LD      A,07H             ; EO,VPP=1 PGM=0
01400         OUT     (PORTC),A         ; TIMING D'ECRITURE PROM
01410 ECRP1   PUSH    DE                ; SAUVEGARDE ADR PROM
01420         LD      A,E               ;
01430         OUT     (PORTA),A         ; ECR POIDS FAIBLE ADR
01440         LD      (3C3CH),A         ; AFFICHAGE ECRAN
01450         SLA     D                 ;
01460         SLA     D                 ;
01470         SLA     D                 ; CADRAGE POIDS FORT ADR
01480         IN      A,(PORTC)         ; LECTURE PORT C
01490         AND     07H               ; PROTECTION BITS PORT C
01500         OR      D                 ; BIT 0-2 + ADR P FORT -> A
01510         OUT     (PORTC),A         ; ECR ADR P FORT PROM
01520         POP     DE                ; RESTITUTION ADR PROM N(3)
01530         LD      A,(HL)            ; DATA A ECRIRE -> A
01540         OUT     (PORTB),A         ; ECRITURE DATA
01550         DI                        ; EVITER LES IT PENDANT 50MS
01560         LD      A,PGM             ; STROB ECR
01570         OUT     (PCNTL),A         ;
01580         PUSH    BC                ; SAUVEGARDE CPT D'OCTET
01590         LD      BC,TIMWP          ; TEMPO ECR PROM
01600         CALL    TEMPO             ; TEMPO 50 MS
01610         LD      A,NPGM            ;
01620         OUT     (PCNTL),A         ;
01630         POP     BC                ; RESTITUTION DE N(4) -> BC
01640         EI                        ;
01650         INC     HL                ; INCREMENT DE N(2)
01660         INC     DE                ; INCREMENT DE N(3)
01670         DEC     BC                ; DECREMENT DE N(4)
01680         LD      A,B               ;
01690         OR      C                 ; TEST SI N(4)=0
01700         JR      NZ,ECRP1          ; SAUT SI <>0
01710         XOR     A                 ; EO,VPP,PGM=0
01720         OUT     (PORTC),A         ; RAZ DU TIMING D'ECRITURE
01730         RET                       ; RETOUR AU BASIC
01740 ;
01750         END     DEBUT
```

```
1 '-----------------------------------'  FF
2 '    PROGRAMME ECRITURE EPROM 2716   '  FE
3 '    (C) TRACE ET JOSUE BONIFAS      '  FD
4 '-----------------------------------'  FC
5 ' Configuration type : modele 1     '  FB
6 '     48 Ko , 1 disquette           '  FA
7 '-----------------------------------'  F9
10 '  F6
20 '--- MENU ---  EC
30 CLS : DEFUSR=(&H8000) : DEFINT N:REM36
40 PRINT @0,"0 - LECTURE 2716":REM05
50 PRINT "1 - COMPARE 2716":REMA3
```



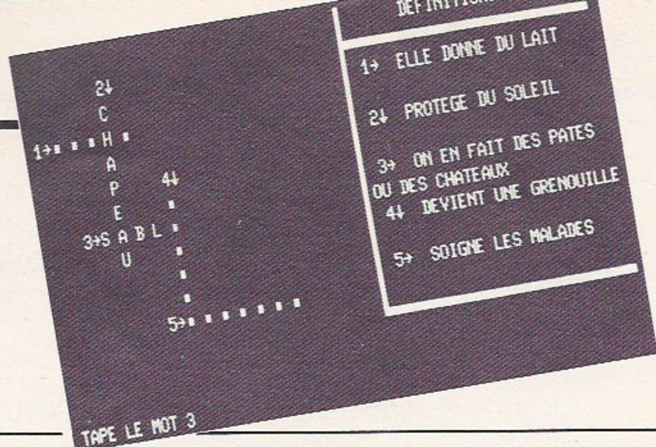
```
60 PRINT "2 - ECRITURE 2716":REM3C
70 PRINT "3 - FIN":REM87
80 INPUT N(0) : IF N(0)>3 THEN 40:REM8B
90 '--- SELECTION ADR DEBUT ET FIN ---  A6
100 CLS : IF N(0)<>0 THEN 150:REM46
110 PRINT @0,"ADRESSE DE LECTURE PROM - DEBUT :":REM43
120 PRINT @88,"- FIN   :":REM72
130 PRINT "ADRESSE  DE  RANGEMENT  MEMOIRE :":REM0F
140 GOTO 240:REM51
150 IF N(0)<>1 THEN 200:REMD5
160 PRINT @0,"ADRESSE DE COMPARAISON PROM - DEBUT :":REME9
170 PRINT @92,"- FIN   :":REM45
180 PRINT "ADRESSE  DE  COMPARAISON  MEMOIRE   :":REM42
190 GOTO 240:REM1F
200 IF N(0)<>2 THEN END:REMB4
210 PRINT @0,"ADRESSE MEMOIRE A ECRIRE - DEBUT :":REMB1
220 PRINT @89,"- FIN   :":REM0D
230 PRINT "ADRESSE  D'ECRITURE   PROM        :":REMD7
240 INPUT A$(2),A$(4),A$(3):REM74
250 '--- CONVERSION EN DECIMAL ---  06
260 FOR I=2 TO 4 : J=0:REMB0
270 FOR I9=1 TO LEN(A$(I)):REME8
280 I9$=MID$(A$(I),I9,1):REM17
290 IF I9$>"@" THEN J9=ASC(I9$)-55 ELSE J9=ASC(I9$)-48:REMBB
300 J=J*16+J9 : NEXT I9:REMA1
310 N(I)=J-INT(J/32768)*65536 : NEXT I:REMC5
320 N(4)=N(4)+1-N(2) : IF N(0)<>2 THEN N(1)=N(2) : N(2)=N(3) : N
(3)=N(1) : GOTO 350:REM31
330 '--- TEST SI COMPTE D'OCTETS <= 2048 ---  B5
340 IF N(4)>2048 THEN CLS : PRINT @25,"ERREUR COMPTE OCTETS" : G
OTO 40:REMBC
350 X=USR(VARPTR(N(0))):REM82
360 '--- AFFICHAGE ERREURS COMPARAISON ---  97
370 IF N(0)<>1 THEN N(0)=1 : GOTO 350:REM56
380 IF N(4)=0 THEN 560:REMB7
390 PRINT "ADR PROM";TAB(17)"DATA";TAB(32)"ADR MEMOIRE";TAB(52)"
DATA":REMF5
400 '--- CONVERSION DECIMAL HEXA ---  6F
410 C$="ABCDEF":REM50
420 FOR I=1 TO 3:REM9B
430 IF SGN(N(I))=-1 THEN R=N(I)+65536 ELSE R=N(I):REM51
440 X=4096:REM47
450 FOR N=0 TO 3:REM79
460 V(N)=INT(R/X) : R=R-V(N)*X : X=X/16:REM33
470 IF V(N)>9 THEN V$(N)=MID$(C$,V(N)-9,1) ELSE V$(N)=RIGHT$(STR
$(V(N)),1):REM0B
480 NEXT:REM98
490 A$(I)=V$(0)+V$(1)+V$(2)+V$(3) : NEXT:REM27
500 PRINT A$(3);TAB(17)RIGHT$(A$(1),2);TAB(32)A$(2);TAB(52)LEFT$
(A$(1),2):REMD7
510 A$=INKEY$ : IF A$=CHR$(13) THEN 560:REMEA
520 IF A$="" THEN 510:REM89
530 N(2)=N(2)+1 : N(3)=N(3)+1 : N(4)=N(4)-1 : IF N(4)=0 THEN 560
:REM0C
540 X=USR(VARPTR(N(0))):REMC3
550 IF N(4)<>0 THEN 420:REM3C
560 GOTO 30:REMDE
```



Josué Bonifas

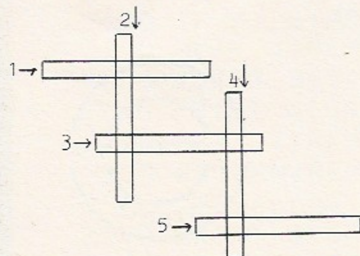
programme

Mots croisés



Ce petit programme a été rédigé initialement pour mes enfants mais il semble pouvoir être utilisé en école primaire : les instituteurs et institutrices qui l'ont vu tourner semblent intéressés... mais attendent d'hypothétiques dotations de matériel...

Ce programme génère des mots croisés au hasard à partir d'un fichier de mots (data à partir de la ligne 10000). Amateurs de Max Favallelli et de Roger de la Ferté : ce programme n'est malheureusement pas pour vous car les mots croisés qu'il crée sont très simples et tous du même type :



La structure de ce programme est la suivante :

Lignes 40-80 : initialisations (en ligne 70, la dimension 50 des tableaux M$,MD$,T correspond au nombre maximum des mots du fichier).
Lignes 100-200 : programme principal qui appelle des sous programmes.
Lignes 1000-1070 : lecture des mots (tableau M$) et de leurs définitions (tableau MD$). Le fichier de mots doit se terminer par le mot "fin" (ligne 1050).
Lignes 1500-1750 : choix du mot croisé. Les variables importantes pour la bonne compréhension de ce sous-programme sont :
- C(i) : rang dans le fichier du ième mot du mot croisé ($1 \leqslant i < 5$).
- P1(i),P2(i) : rang des croisements du ième mot avec les autres mots (P1(i) < P2(i)).
- T(j) : "drapeau" (T(j) = 1 si le jème mot du fichier est pris dans le mot croisé, 0 sinon).

Lignes 1800-1870 : ce sous programme sert à ne pas avoir d'interférence entre les mots 1 et 4, 2 et 5 et à ce que le mot croisé n'empiète pas sur la partie définition des mots à l'écran.
Lignes 2000-2230 : affichage de la grille à gauche de l'écran (numéro du mot suivi de → pour un mot horizontal, de ↓ pour un mot vertical et des pavés graphiques pour les lettres) et affichage des définitions des mots à droite de l'écran.
Lignes 2500-4160 : réponse de l'enfant avec test et tout ce qu'il faut (simulation d'un INPUT par INKEY$, possibilité d'effacer une lettre mal introduite).
Lignes 4500-4570 : le mot croisé est fini : est-ce qu'on continue ?

Pour bien utiliser ce programme, il suffit normalement de bien lire ce qui est affiché à l'écran mais les enfants ont parfois du mal. Alors précisons :

1. Il faut toujours introduire le numéro du mot auquel on pense avant d'introduire le mot (même quand il ne reste plus qu'un mot à trouver).
2. L'introduction d'un mot doit toujours se terminer par la frappe de ENTER, comme s'il s'agissait d'une instruction INPUT. Les lettres d'un mot trouvé restent affichées, sinon elles sont effacées.
3. Lorsqu'on se trompe dans la frappe d'un mot, on peut revenir en arrière en tapant sur les touches :

← pour un mot horizontal
↑ pour un mot vertical

Les lettres frappées sont affichées sur la grille à la place du curseur clignotant.

4. Lorsqu'un mot a été trouvé, ses lettres lorsqu'elles appartiennent à un autre mot, ne sont plus à taper. C'est normal pour un mot croisé, mais peu habituel pour un utilisateur d'ordinateur. L'utilisation des touches ← et ↑ tient compte de cela. De toute façon, le curseur clignotant est, à mon avis, très explicite.
5. Le curseur clignotant n'est plus visible dans deux cas, mais c'est bien normal :
   a) On a tapé autant de lettres que le mot cherché en comporte. Dans ce cas, l'ordinateur attend pour commande : ← ou ↑ (rectification de lettres) ou ENTER (fin de l'introduction d'un mot).
   b) La dernière lettre du mot cherché est connue grâce à un autre mot. Comme cette dernière lettre n'est plus à taper, à la frappe de l'avant dernière lettre, on se retrouve dans la situation du a).

Pour tenir compte des mots déjà trouvés, j'ai utilisé une astuce évitant des recherches fastidieuses dans des tableaux. Pour un mot trouvé, à côté de chaque lettre du mot est affiché le caractère graphique 128 (alors que pour un mot non trouvé c'est le caractère blanc codé 32 qui est affiché !). Un PEEK de l'écran permet au programme de s'y retrouver : l'utilisateur, lui, ne voit rien (lignes 3030, 3420, 4050).

Ces quelques indications doivent suffire pour comprendre le fonctionnement de ce programme et bien l'utiliser.
Pour une utilisation pédagogique, il faut à mon avis quelques modifications et il convient de respecter certaines contraintes :

1. Création de fichiers de mots (difficultés orthographiques d'un certain type, mots concernant un thème...). Selon que vous possédez cassettes ou disquettes, la gestion de tels fichiers étant très différente, je vous laisse l'adapter à votre matériel.
2. Contrainte de longueurs. Les mots doivent avoir 5, 6 ou 7 lettres (la hauteur maximale occupée par les mots 2 et 4 est de 14 lignes). Les définitions ont au plus 51 caractères (contrainte due à la zone réservée aux définitions et comme les mots d'une définition ne sont pas coupés, il faut prévoir un peu moins).
3. Importance des fichiers. Les fichiers doivent comprendre au minimum une trentaine de mots pour permettre une certaine diversité. Il faut aussi éviter un trop grand nombre de mots commençant et finissant par "e" : ils risquent de revenir très souvent.

Si certains lecteurs utilisent ce programme en classe, qu'ils nous fassent part de leurs réactions, de celles des enfants... et de leurs suggestions qui seront les bienvenues.

```
1 '          Mots-croises                    '  FF
2 ' (C) Gerald Grandpierre et TRACE         '  FE
3 ' Configuration : modele 1 ou 3, 16 K     '  FD
4 '--------------------------------------------  FC
40 CLS:REM54
50 DEFINT A-Z:REMCC
60 CLEAR 5000:REM47
70 DIM M$(50),MD$(50),C(5),T(50),P1(5),P2(5),EC(5):REMA6
80 RANDOM:REM2A
100 ' Programme principal  9C
110 GOSUB 1000:REM4D
120 GOSUB 1500 : GOSUB 1800 : IF F=1 THEN 120:REM2B
130 GOSUB 2000:REM2B
150 GOSUB 2500:REM12
160 GOSUB 4000:REM0B
170 IF BO<>5 THEN 150:REMF7
180 GOSUB 4500:REMF2
190 IF R$="OUI" THEN CLS : GOTO 120:REM8F
200 END:REMB8
1000 ' Lecture des mots et des definitions  15
1010 N=0:REMB8
1020 RESTORE:REM71
1030 N=N+1:REM87
1040 READ M$(N):REM51
1050 IF M$(N)="FIN" THEN N=N-1 : RETURN:REM47
1060 READ MD$(N)
1070 GOTO 1030:REM7D
1500 ' Choix des mots  1F
1510 FOR I=1 TO N:REM3A
1520 T(I)=0:REM18
1530 NEXT:REM7A
1540 FOR I=1 TO 5:REM34
1550 P1(I)=0 : P2(I)=0:REM71
1560 NEXT:REM5B
1600 F=0:C(1)=RND(N) : T(C(1))=1:REM74
1610 FOR I=2 TO 5:REMED
1620 C(I)=RND(N) : IF T(C(I))=1 THEN 1620 ELSE T(C(I))=1:REM3C
1630    FOR J=0 TO 2:REMDD
1640    C$=MID$(M$(C(I-1)),LEN(M$(C(I-1)))-J,1):REMEA
1650    B=3:IF LEN(M$(C(I)))-3<B THEN B=LEN(M$(C(I)))-3:REMB4
1660        FOR K=1 TO B:REMAD
1670        IF C$=MID$(M$(C(I)),K,1) THEN 1710:REM58
1680        NEXT K:REM98
1690    NEXT J:REM8F
1700 T(C(I))=0 : GOTO 1620:REM3F
1710 P1(I)=K : P2(I-1)=LEN(M$(C(I-1)))-J:REMED
1720 IF P2(I-1)-P1(I-1)=1 THEN 1700:REM18
1730 NEXT I:REM68
1740 P2(0)=P1(2):REM52
1750 RETURN:REM92
1800 ' Cas a exclure  F1
1810 IF P2(3)-P1(3)-1>LEN(M$(C(3)))-P2(1) OR P2(2)-P1(2)>P1(4) T
HEN 1830:REMF3
1820 F=1 : RETURN:REMC5
1830 IF P2(3)-P1(3)>P1(5) OR P2(4)-P1(4)-1>LEN(M$(C(2)))-P1(2) T
HEN 1850:REMD9
1840 F=1 : RETURN:REMB1
1850 IF LEN(M$(C(5)))-P1(5)+P2(3)-P1(3)+P2(1)<18 THEN 1870:REM3C
```



```
1860 F=1:REM69
1870 RETURN:REM19
2000 ' Affichage  29
2010 CLS:REM9B
2020 FOR I=0 TO 37 : SET(72,I) : NEXT:REM72
2030 PRINT @43,"DEFINITIONS";:REMCB
2040 FOR I=73 TO 127 : SET(I,4) : SET(I,37) : NEXT:REM3C
2050 EC=64:REM2F
2060 FOR I=0 TO 4:REM2C
2070 IF I/2=INT(I/2) THEN U=64 : V=2 : D$=CHR$(94) ELSE U=2 : V=
64 : D$=CHR$(92):REM7A
2080 EC=EC+P2(I)*U-P1(I+1)*V : EC(I+1)=EC+V:REM00
2100 PRINT @EC,RIGHT$(STR$(I+1),1);D$; : PRINT @38+128*(I+1),RIG
HT$(STR$(I+1),1);D$;:REM6A
2110 FOR J=1 TO LEN(M$(C(I+1))):REM4B
2120 PRINT @EC+J*V,CHR$(136);:REMB1
2130 NEXT J:REMD5
2150 L=LEN(MD$(C(I+1))):REM4C
2160 IF L<23 THEN PRINT @42+128*(I+1),MD$(C(I+1)); : GOTO 2220:R
EMAC
2170 FOR J=23 TO 1 STEP -1:REMC0
2180 IF MID$(MD$(C(I+1)),J,1)=" " THEN 2200:REM3F
2190 NEXT J:REM99
2200 PRINT @42+128*(I+1),LEFT$(MD$(C(I+1)),J-1);:REM33
2210 PRINT @101+128*(I+1),RIGHT$(MD$(C(I+1)),L-J);:REME1
2220 NEXT I:REM7C
2230 RETURN:REMB0
2500 ' Reponse de l'enfant  33
2510 PRINT @960,"A QUEL MOT PENSES-TU (";:REMB9
2520 FOR I=1 TO 5:REM5D
2530 IF C(I)<>0 THEN PRINT I;" ";:REM50
2540 NEXT : PRINT ") ?";:REMB1
2550 A$=INKEY$ : IF A$="" THEN 2550:REM21
2560 CH=VAL(A$) : IF C(CH)=0 THEN 2550:REM68
2570 PRINT @960,STRING$(60,32);:REME8
2580 PRINT @960,"TAPE LE MOT";CH;:REM35
2590 GOSUB 3000:REM84
2600 RETURN:REM3C
3000 ' Introduction d'un mot  3D
3010 EM=EC(CH) : R$="" : K=136 : L=LEN(M$(C(CH))):REM2B
3020 IF CH-2*INT(CH/2)=0 THEN U=64 ELSE U=2:REMFC
3030 IF PEEK(15361+EM)=128 THEN R$=R$+CHR$(PEEK(15360+EM)) : EM=
EM+U : GOTO 3030:REM75
3040 A$=INKEY$ : IF A$="" THEN GOSUB 3200 : GOTO 3040:REM1D
3050 IF ASC(A$)=13 AND LEN(R$)=L THEN RETURN:REM8E
3060 IF ((U=2 AND ASC(A$)=8) OR (U=64 AND ASC(A$)=91)) AND LEN(R
$)>0 THEN GOSUB 3400 : GOTO 3040:REM34
3070 IF A$<"A" OR A$>"Z" THEN 3040:REM6D
3080 IF LEN(R$)=L THEN 3040:REMF1
3090 R$=R$+A$ : PRINT @EM,A$; : EM=EM+U:REM10
3100 GOTO 3030:REM85
3200 ' Attente  74
3210 IF LEN(R$)=L THEN RETURN:REMA4
3220 PRINT @EM,CHR$(K);:REME2
3230 FOR I=1 TO 20 : NEXT:REMA6
3240 K=168-K:REM74
3250 RETURN:REMB0
3400 ' Effacage d'un caractere  AB
3410 IF LEN(R$)<>L THEN PRINT @EM,CHR$(136);:REMCB
3420 IF PEEK(15361+EM-U)=32 THEN EM=EM-U : R$=LEFT$(R$,LEN(R$)-1
) : RETURN:REM3B
```



```
3430 IF LEN(R$)>1 THEN EM=EM-2*U : R$=LEFT$(R$,LEN(R$)-2):REM2E
3440 RETURN:REMF1
4000 ' Test de la reponse  51
4010 IF R$=M$(C(CH)) THEN 4100:REMFD
4020 PRINT @960,"CE N'EST PAS CE MOT";:REMBC
4030 FOR I=1 TO 2000 : NEXT:REM23
4040 FOR I=0 TO L-1:REM52
4050 IF PEEK(15361+EC(CH)+I*U)=128 THEN 4070:REMEA
4060 PRINT @EC(CH)+I*U,CHR$(136);:REM3C
4070 NEXT:REM84
4080 RETURN:REM6F
4100 PRINT @960,"C'EST BIEN CE MOT";:REMC4
4110 FOR I=1 TO 2000 : NEXT:REMD2
4120 FOR I=0 TO L-1:REM01
4130 PRINT @EC(CH)+I*U+1,CHR$(128);:REMF6
4140 NEXT:REM3D
4150 BO=BO+1 : C(CH)=0:REM67
4160 RETURN:REM1E
4500 ' Mot croise termine  5B
4510 PRINT @960,"BRAVO,TU AS TROUVE TOUS LES MOTS";:REMDB
4520 FOR I=1 TO 2000 : NEXT:REM37
4530 PRINT @960,"VEUX-TU FAIRE D'AUTRES MOTS CROISES";:REME2
4540 INPUT R$:REM34
4550 IF R$<>"OUI" AND R$<>"NON" THEN PRINT@960,STRING$(63,32); :
 GOTO 4530:REMC4
4560 BO=0:REM89
4570 RETURN:REM83
10000 ' Mots et definitions  C9
10010 DATA VACHE,ELLE DONNE DU LAIT:REM4B
10020 DATA SAPIN,ON LE DECORE A NOEL:REM37
10030 DATA COUTEAU,SERT A COUPER:REM94
10040 DATA COCHON,A LA QUEUE EN TIRE BOUCHON:REM0B
10050 DATA POISSON,VIT DANS L'EAU:REM51
10060 DATA ETOILE,BRILLE LA NUIT DANS LE CIEL:REMBC
10070 DATA AVION,VOLE DANS LE CIEL:REM48
10080 DATA SABLE,ON EN FAIT DES PATES OU DES CHATEAUX:REMDE
10090 DATA BONBON,DEUX FOIS BON:REMB7
10100 DATA ECURIE,MAISON DU CHEVAL:REME1
10110 DATA BOUCHE,SERT POUR MANGER ET PARLER:REM54
10120 DATA BIBERON,SERT A FAIRE BOIRE LES BEBES:REM00
10130 DATA ARBRE,A UN TRONC ET DES BRANCHES:REM02
10140 DATA ETABLE,LES VACHES Y HABITENT:REM96
10150 DATA ECOLE,ELLE EST FERMEE PENDANT LES VACANCES:REM23
10160 DATA BOTTE,CHAUSSURE POUR LA PLUIE:REMF2
10170 DATA GIRAFE,ANIMAL AU LONG COU:REM5E
10180 DATA SAVON,SERT A BIEN SE LAVER:REM2B
10190 DATA ECHELLE,SERT A MONTER DANS LES ARBRES:REM48
10200 DATA ROUGE,COULEUR DU SANG:REMEA
10210 DATA JAUNE,COULEUR DU CITRON:REM49
10220 DATA CHAPEAU,PROTEGE DU SOLEIL:REMCB
10230 DATA DOCTEUR,SOIGNE LES MALADES:REM79
10240 DATA SERPENT,EST PARFOIS A SONNETTE:REM52
10250 DATA RUCHE,LES ABEILLES Y FONT LE MIEL:REM36
10260 DATA TETARD,DEVIENT UNE GRENOUILLE
10270 DATA TOBOGAN,ON Y GLISSE:REM3F
10280 DATA MARTEAU,SERT A ENFONCER DES CLOUS:REMBC
10290 DATA TABLEAU,ON Y ECRIT AVEC UNE CRAIE:REM1C
10300 DATA CHAISE,ON S'Y ASSOIT:REMF8
10310 DATA PINCEAU,SERT A PEINDRE:REM53
11000 DATA FIN:REM79
```



Gérald Grandpierre

Copie de fichiers sur 1 drive

TRACESYS est destiné à permettre aux utilisateurs de TRS80 Modèle 1, qui ne disposent que d'une seule unité de disquette et de TRSDOS, de recopier néanmoins les fichiers des disquettes TRACE sur des disquettes système. En effet, la commande COPY des TRSDOS versions 2.1 à 2.3 nécessite deux unités car la copie se fait secteur par secteur.

Afin que ceux qui ne possèdent pas d'assembleur puissent également en disposer, le programme a été écrit en BASIC. Mais oui, ce langage mène à tout à condition d'en sortir parfois grâce à l'instruction USR ! Il y aura donc des DATA pour alimenter le POKE de la boucle d'implantation de quatre petites routines en langage machine. Celles-ci sont réduites au strict minimum (1Ø9 octets en tout seulement), pour soulager le pianiste tapant les fastidieux DATA. Il en résulte évidemment certaines limitations qui auraient pû être évitées avec un programme plus sophistiqué mais plus volumineux. Tel quel, outre l'emploi dans sa vocation initiale, vous pourrez également l'utiliser pour la copie de fichiers d'une disquette non système sur une disquette système, quel que soit le DOS (en principe du moins). Il fonctionne également sur le TRS80 Modèle 3, mais la disquette source doit avoir été générée par le DOS de la disquette destination, en raison de la gestion différente de la DIRECTORY. La taille de chacun des fichiers ne devra pas excéder 16 ou 32 K octets selon que vous disposez de 32 ou 48 K de MEV, sinon vous écraserez la pile et ce sera le plantage. D'autre part, lors de la copie de fichier ayant une longueur d'enregistrement logique (LRL) différente de 256 octets, le nouveau fichier sera créé avec une longueur d'enregistrement de 256 octets (LRL=ØØ). Pour obtenir une copie conforme, il faut créer préalablement le fichier destination (à l'aide de la commande CREATE par exemple) avec la LRL désirée.

Pour la mise en oeuvre, respectez le mode opératoire suivant :

- Placez la disquette système TRSDOS contenant le programme TRACESYS dans l'unité 0 et chargez-la.
- Appelez Basic et tapez RUN"TRACESYS".
- Pour annuler une opération en cours (hors des accès disque) et revenir au début de la procédure de copie, pressez la touche (A).
- Appuyez sur (Q) pour terminer la session et retourner au Basic.
- Ne jamais faire RUN tout seul à la suite d'une commande SAVE ou après l'affichage d'erreur, car alors le module système en overlay ne serait pas SYS2/SYS, lequel est absolument nécessaire pour débuter la procédure avec la phase de lecture.

La séquence de copie d'un fichier avec TRACESYS demande beaucoup moins de manipulations de disquettes que la plupart des utilitaires fonctionnant avec une seule unité :

- Après avoir déclaré le nom du fichier, l'usager est invité à placer la disquette TRACE dans le lecteur. Le chargement en mémoire dans un tampon, s'effectue en une seule passe.
- Ensuite il faut insérer la disquette destination qui doit obligatoirement être une disquette système. Un nouvel accès disque réalisera en une seule fois toutes les opérations d'écriture du fichier. Si le mode VERIFY a été activé, chaque secteur sera relu immédiatement au tour de disque qui suit celui de son écriture.
- Normalement la séquence se termine avec l'affichage d'un message indiquant que la copie a été effectuée et invitant à continuer. En cas d'anomalie rencontrée lors de l'écriture, un message d'erreur envoyé par le DOS sera affiché à sa place. Dans ce cas, vous êtes invité à placer la disquette système dans l'unité 0. Gageons que le cas le plus fréquent sera, par suite d'une erreur de frappe, "FILE NOT IN DIRECTORY..."

```
1 ' Recopie des fichiers de la disquette TRACE        ' FF
2 ' avec TRS-DOS et une seule unite de disquette      ' FE
3 '            sur modele 1 ou VGS                    ' FD
4 '                                                   ' FC
5 ' (C) 1983 - Roger Buvat et TRACE                   ' FB
6 ' ================================================= FA
7 '  F9
10 CLS : PRINT"
COPIE DES FICHIERS DE DISQUETTE 'TRACE' SUR DISQUETTE SYSTEME":R
EMC5
20 PRINT"AVEC LE DRIVE 0 SOUS TRSDOS (TRS-80 MODELE 1)":REMAD
30 M=32000 : REM START=7D00H  CB
40 FOR I=M TO M+108 : READ J : POKE I,J : NEXT:REM02
50 DEFUSR0=M : DEFUSR1=M+16 : DEFUSR2=M+48 : DEFUSR3=M+96:REM65
60 A$="PRESSER <ENTER>" : B$="POUR CONTINUER ":REM74
70 PRINT : PRINT : PRINT"ENTRER LE NOM DU FICHIER : ";:REM99
80 X=USR0(0) : REM PLACE NOMFICH DANS FCB  11
90 PRINT : E$="'TRACE'" : GOSUB210:REM1D
100 X=USR1(0) : REM CHARGE LE FICHIER  FC
```



```
110 IF X<>0 GOTO170 : REM SI ERREUR A LA LECTURE  AC
120 E$="DESTINATION" : GOSUB210:REM66
130 X=USR2(0) : REM ECRIT LE FICHIER  DD
140 IF X<>0 GOTO180 : REM SI ERREUR A L'ECRITURE  8D
150 PRINT"COPIE FAITE. ";B$;A$ : GOSUB220:REMCD
160 GOTO70:REM6C
170 E$="SYSTEME" : GOSUB210:REM4C
180 X=USR3(X) : REM AFFICHE LE MESSAGE D'ERREUR  82
190 PRINT : PRINT B$;A$ : GOSUB220:REM3F
200 RUN"TRACESYS":REMF8
210 PRINT"PLACER LA DISQUETTE ";E$;" DANS LE DRIVE ET ";A$:REMAF
220 K$=INKEY$ : IF K$="" GOTO220:REMA5
230 IF K$="A" GOTO70:REMCE
240 IF K$="Q" END:REM28
250 IF ASC(K$)<>13 GOTO220 : REM SI NON <ENTER>  92
260 PRINT : RETURN:REM7D
270 DATA 33,109,125,6,24,205,64,0,17,141:REM7A
280 DATA 125,6,0,237,176,201,33,173,125,17:REM07
290 DATA 109,125,6,0,205,36,68,32,76,205:REM5A
300 DATA 54,68,32,6,33,113,125,52,24,245:REM52
310 DATA 254,28,200,254,29,200,24,57,33,173:REMB3
320 DATA 125,17,141,125,6,0,205,32,68,32:REM48
330 DATA 39,205,57,68,32,34,33,145,125,52:REMF9
340 DATA 58,113,125,190,32,241,58,117,125,50:REM64
350 DATA 149,125,42,121,125,34,153,125,205,40:REM32
360 DATA 68,32,7,195,36,68,205,127,10,125:REMD9
370 DATA 246,192,205,9,68,111,195,154,10:REM02
```



Roger Buvat

utilitaire

Un complément à EDTASM version cassette

Cet utilitaire complémentaire à EDTASM (d'où son nom CETASM) permet de sortir d'EDTASM, de travailler en Basic et de revenir dans EDTASM sans rechargement et sans perte des programmes source et Basic.
Pour les utilisateurs d'un modèle 3 à cassette, CETASM permet en outre de charger EDTASM à 1500 bauds, ce qui est plus agréable.

Avant de voir la réalisation pratique, voici le principe de CETASM.
A son lancement, ce programme déplace l'origine du tampon Basic pour laisser le début de la MEV disponible pour EDTASM. Cette zone est alors automatiquement protégée et CETASM se place à sa suite.
L'origine de CETASM doit être de préférence une adresse simple car elle sera sollicitée à chaque retour dans EDTASM. J'ai personnellement choisi 30000 mais elle peut varier (de 27000 à 60000) selon la taille mémoire, l'importance relative du programme source et celle de la zone Basic.
Après le déplacement du tampon Basic qui s'implante immédiatement après CETASM, l'utilitaire charge EDTASM à 1500 bauds, revient à 500 bauds, et lance ce dernier après l'avoir modifié, pour éviter la "raz" au moment de la sortie. En fait, cette modification permet de revenir dans CETASM (à la ligne 450) qui terminera la sortie d'EDTASM par un effacement de l'écran et la question "Cass?".
Le retour dans EDTASM se fait à partir du Basic par la commande SYSTEM en passant par CETASM qui rétablit la vitesse à 500 bauds.

Passons maintenant à la réalisation.
Charger EDTASM à 1500 bauds signifie lire le programme à 1500 bauds. Avant cela, il faut donc avoir écrit ce programme sur cassette à cette vitesse. C'est l'objet du programme ECRI qui est très simple.
La constitution de la cassette CETASM-EDTASM rapide peut se faire en deux temps. D'abord avec l'aide d'EDTASM (lent), entrez, puis assemblez CETASM et sauvez-le sur la cassette en prévoyant à la suite une place nécessaire à EDTASM (à titre indicatif, CETASM + EDTASM occupent au compteur les graduations 3 à 24).
Entrez et assemblez également ECRI et sauvez-le sur une autre cassette.
Enchaînez ensuite les opérations suivantes, après avoir "poké" 0 dans les mémoires 17500 à 26000 (simple précaution) :

Faites une raz :	Cass? L
	....
- puis :	>SYSTEM
- chargez EDTASM /	*? EDTASM

- chargez ECRI : *? ECRI
- lancez ECRI (sans adresse) : *? /
  READY
- continuez : >SYSTEM
- chargez CETASM : *? CETASM
- appuyez sur la touche Enter : *?

ce qui doit donner un message d'erreur. Vous êtes en Basic et vous avez localisé la fin de votre programme CETASM sur la cassette,

- revenez à >SYSTEM
- appuyez sur RECORD et PLAY
- lancez ECRI : *? /31000

Pour ce qui est de l'utilisation, c'est très simple.

- faites une raz : Cass? L
- réservez une place en fond de mémoire pour votre programme assemblé : Memory Size? XXXXX
  ....
- entrez : >SYSTEM
- chargez CETASM : *? CETASM
- lancez CETASM (sans adresse) : *? /

et laissez faire...

Vous pouvez quitter EDTASM normalement par la commande Q,

- choisir votre vitesse : Cass? H
  et travailler en Basic.

Lorsque vous voulez revenir dans EDTASM, faites : SYSTEM /30000, c'est-à- dire l'adresse origine de CETASM.

Nous venons de voir la marche à suivre pour un système Modèle 3 cassette. Dans le cas d'un Modèle 1, le principe de CETASM peut, je pense, être utilisé en supprimant bien sûr, les lignes 120, 130, 200 à 400 relatives à la vitesse de lecture. Il y a lieu de remplacer l'adresse OFCH (retour au Basic) par 6CCH.

Vous devez ensuite charger par : >SYSTEM
successivement, EDTASM : *? EDTASM
puis CETASM : *? CETASM
avant de lancer l'exécution : *? /

```
               00010 ;-------------------------------
               00020 ;            ECRI
               00030 ;
               00040 ; (C) 1983 - TRACE et l'auteur
               00050 ;-------------------------------
7918           00060          ORG     31000
               00070 ;
7918 3E0D      00080          LD      A,0DH
791A 321142    00090          LD      (4211H),A
791D 3E00      00100          LD      A,0
791F CD1202    00110          CALL    212H
7922 CD8702    00120          CALL    287H
7925 214646    00130          LD      HL,4646H
7928 061D      00140          LD      B,1DH
792A C5        00150 SUI      PUSH    BC
792B 06FF      00160          LD      B,0FFH
792D 7E        00170 ECR      LD      A,(HL)
792E CD6402    00180          CALL    264H
7931 23        00190          INC     HL
7932 10F9      00200          DJNZ    ECR
7934 C1        00210          POP     BC
7935 10F3      00220          DJNZ    SUI
7937 CDF801    00230          CALL    1F8H
               00240 ;
793A AF        00250          XOR     A
793B 321142    00260          LD      (4211H),A
               00270 ;
793E CDC901    00280 SORTIE   CALL    1C9H
7941 C3FC00    00290          JP      0FCH
               00300 ;
793E           00310          END     SORTIE
00000 TOTAL ERRORS
```

```
               00010 ;-----------------------------
               00020 ;           CETASM
               00030 ;
               00040 ; (C) 1983 - TRACE et l'auteur
               00050 ;-----------------------------
7530           00060          ORG     30000
7530 AF        00070          XOR     A
7531 321142    00080          LD      (4211H),A     ; Cass 500 bds
7534 C30E4C    00090          JP      4C0EH
               00100 ;
7537 217A75    00110 DEBUT    LD      HL,BUFBAS     ; Deplacement
753A 22A440    00120          LD      (40A4H),HL    ; Basic
753D CD4D1B    00130          CALL    1B4DH         ; NEW
               00140 ;
7540 3E0D      00150          LD      A,0DH
7542 321142    00160          LD      (4211H),A     ; Cass 1500 bds
               00170 ;
7545 3E00      00180          LD      A,0
7547 CD1202    00190          CALL    212H
754A CD9602    00200          CALL    296H
754D 214646    00210          LD      HL,4646H
7550 061D      00220          LD      B,1DH
7552 C5        00230 SUI      PUSH    BC
7553 06FF      00240          LD      B,0FFH
7555 CD3502    00250 LEC      CALL    235H
7558 77        00260          LD      (HL),A        ; Lecture cassette
7559 23        00270          INC     HL
755A 10F9      00280          DJNZ    LEC
755C C1        00290          POP     BC
755D 10F3      00300          DJNZ    SUI
755F CDF801    00310          CALL    1F8H
               00320 ;
7562 AF        00330          XOR     A
7563 321142    00340          LD      (4211H),A     ; Cass 500 bds
               00350 ;
7566 216F75    00360          LD      HL,ADRESS
7569 22F54F    00370          LD      (4FF5H),HL
756C C3D34B    00380          JP      4BD3H
               00390 ;
756F CDC901    00400 ADRESS   CALL    1C9H          ; CLS
7572 CD4230    00410          CALL    3042H         ; Cass?
7575 C3FC00    00420          JP      0FCH          ; Retour Basic
               00430 ;
7578 0000      00440          DEFW    0
757A 0000      00450 BUFBAS   DEFW    0
757C 0000      00460          DEFW    0
757E 0000      00470          DEFW    0
               00480 ;
7537           00490          END     DEBUT
00000 TOTAL ERRORS
```

Camille Bochard

CUBUS: un jeu pour TRS couleur

Il s'agit ici du premier programme que nous publions sur le TRS couleur, à la demande de quelques utilisateurs de ce matériel. Espérons que ce ne sera pas le dernier...! (Ndlr)

Le principe de ce jeu est très simple, il vous suffit d'actionner la manette à droite ou à gauche pour vous déplacer. Une petite ligne orange qui sert de réceptacle indique que vous devez attraper les "météorites" qui tombent du "ciel".
Si vous désirez vous déplacer deux fois plus vite, appuyez sur le bouton rouge de la manette. Le score est affiché dans le coin supérieur gauche et dans le coin supérieur droit est inscrit le nombre de parties restantes (il y en a trois).

Vous perdez une partie si :

1) vous touchez l'un des deux piliers rouges ;
2) une "météroïte" touche les fondations (la ligne rouge au bas de l'écran) ;
3) tous les cubes sont abattus.

Matériel nécessaire : TRS couleur (étendu) + manette de jeu.

```
1 '+++++++++++++++++++++++++++++
2 '+    cubus : JEU D'ADRESSE  +
3 '+                           +
4 '+MAT. : TRS COULEUR(ETENDU) +
5 '+                           +
6 '+  (C) WAUTELET YANNICK     +
7 '+++++++++++++++++++++++++++++
50 YY=1 : G=0
60 F=1
70 CLS(0)
80 FOR H = 0 TO 1 : FOR V= 19 TO 30 : SE
T(H,V,4) : NEXTV,H
90 FOR H= 62 TO 63 : FOR V= 19 TO 30 : S
ET(H,V,4) : NEXT V,H
100 FOR H= 0 TO 4 : FOR V= 19 TO 21 : SE
T(H,V,4) : NEXTV,H
110 FOR H=58 TO 63 : FOR V = 19 TO 21 :
SET(H,V,4) : NEXTV,H
120 FOR V = 0 TO 63 : SET(V,30,4): NEXT
130 A=4 : B=8 :C=1:A1=4:A2=8:A3=12:A4=16
:A5=20:A6=24:A7=28:A8=32:B1=36:B2=40:B3=
44:B4=48:B5=52:B6=56:K=1:L=30:R=2:TT=1
140 FOR H = A TO B
150 FOR V = 26 TO 29
160 SET(H,V,C)
170 NEXT V,H
180 A=A+8 : B=B+8 : C=C+1
190 IF A = 60 THEN PRINT@455,"c";:PRINT@
459,"u"; :PRINT@463,"b";: PRINT@467,"u";
 : PRINT@471,"s"; : GOTO 210
200 GOTO 140
210 D=(RND(7)*8)-2 : E=RND(7)
220 PRINT@1,G; : IF YY = 1 THEN PRINT@29
,"x  ";
230 IF YY=4 THEN GOTO 490
240 IF YY= 2 THEN PRINT@29,"xx ";
250 IF YY = 3 THEN PRINT@29,"xxx";
260 F=F+1
270 RESET(D,F-1)
280 IF F=30 THEN :PLAY"L255;A;B;C;A;B;C;
A;B;C" : GOTO 480
290 IF POINT(D,F) = 8 THEN PLAY"L255;A;B
;C;D;E;F;G" : G=G+1 : PRINT@1,G; : F=1 :
 GOTO 210
300 IF POINT(D,F) = 1 THEN TT=TT+1 : I=A
1 : J=A2 : GOTO 460
310 IF POINT(D,F) = 2 THEN TT=TT+1 : I=A
3 : J=A4 : GOTO 460
320 IF POINT(D,F) = 3 THEN TT=TT+1 :  I=
A5 : J=A6 : GOTO 460
330 IF POINT(D,F) = 4 THEN TT=TT+1 :  I=
A7 : J=A8 : GOTO 460
340 IF POINT(D,F) = 5 THEN I=B1 : J=B2 :
 GOTO 460
350 IF POINT(D,F) = 6 THEN TT=TT+1 : I=B
3 : J=B4 : GOTO 460
360 IF POINT(D,F) = 7 THEN TT=TT+1 :  I=
B5 : J=B6 : GOTO 460
370 SET(D,F,E)
380 IF TT = 7 THEN GOTO 480
390 X=JOYSTK(0) : Y=JOYSTK(1) : P=PEEK(6
5280) : IF P = 126 OR P=254 THEN K=2 : R
=3 ELSE K=1 : R=2
400 IF X < 20 THEN L=L-K : RESET(L+K,20)
 : RESET(L+R,20)
410 IF X > 20 THEN L=L+K : RESET(L-K,20)
 : RESET(L-R,20)
420 IF L=4 OR L=3 OR L=5 THEN FOR H = 1
TO 20: PLAY"L255;G" :  SET(RND(8),RND(7)
+16,RND(8)) : NEXT : GOTO 480
430 IF L=58 OR L=59 OR L=57 THEN FOR H=1
 TO 20 : PLAY"L255;G" : SET(RND(6)+53,RN
D(7)+16,RND(8)) : NEXT : GOTO 480
440 SET(L,20,8) : SET(L+1,20,8)
450 GOTO 260
460 F=1 : FOR Y = 1 TO 20:RESET(RND(J-I)
+I-1,RND(4)+25) : PLAY"L255;B;C" : NEXT
Y : FOR H = I TO J : FOR V= 26 TO 29 : R
ESET(H,V) : NEXT V,H : I=0 : J=0
470 GOTO 210
480 YY=YY+1 : GOTO 60
490 PRINT@511,""; : FOR A = 1 TO 15 : PR
INTSTRING$(31," ") : FOR S = 1 TO 40 : N
EXT S : NEXT A
500 PRINT@173,"cubus"
510 PRINT@260,"VOTRE SCORE :";G
520 PRINT@449,"VOULEZ VOUS CONTINUEZ (O-
N)"; : INPUTA$
530 IF A$ = "O" THEN GOTO 50
540 END
```

Yannick Wautelet

Un traitement de texte très beau, pas cher

Le programme de traitement de texte publié initialement dans TRACE numéro 5 et amélioré dans TRACE numéro 6 a fait couler beaucoup d'encre et a rencontré un vif succès si l'on en juge par le courrier reçu... toujours écrit à l'aide de ce programme ! De nombreux lecteurs nous ont proposé des améliorations et nous tenons à les féliciter et à les remercier pour leur contribution, tout particulièrement Monsieur Maurice Lhomme (Oullins). Parallèlement, Roger Brousmiche, l'auteur du programme, continue à faire évoluer son "bébé" et nous soumet régulièrement des corrections et améliorations qui, la plupart du temps, englobent celles des autres lecteurs. Nous vous faisons bénéficier des dernières grâce auxquelles le texte peut maintenant être justifié à droite. Les lecteurs recevant la disquette TRACE trouveront naturellement la version intégrale du programme MACHEC incluant ces dernières améliorations. (Ndlr)

La présentation d'un texte imprimé est nettement plus agréable lorsque la marge droite est constante. L'alignement automatique de la marge droite est maintenant chose courante dans les logiciels de traitement de texte du commerce. Il est obtenu par le report de quelques blancs de fin de ligne dans les espaces déjà présents à l'intérieur de la ligne (les premiers espaces sont doublés). Cette amélioration peut être implantée à peu de frais au 'mini" traitement de texte décrit dans les numéros 5 et 6 de TRACE.

La ligne 400 est modifiée pour l'appel de la sous-routine 2000 à 2030 de reformatage de la ligne, après la frappe de 'ENTER'.
A la ligne 50, le 'bip' sonore est déclenché à JL-5 (JL = nombre de caractères par ligne), pour signaler que les caractères suivants seront justifiés à droite.

La modification de la ligne 460 empêche la confusion du texte à l'écran lorsque la ligne contient plus de 63 caractères.

Les modifications des lignes 80, 150, 160 et 180 permettent le déplacement du curseur (flèche haute ou basse), dans les lignes de plus de 63 caractères. Il ne faut cependant pas perdre de vue que la position courante du curseur définit toujours la fin de ligne lorsqu'on frappe 'ENTER'.

A la ligne 10, la variable ME (nombre de lignes maximum par page), est définie différemment. On peut maintenant mémoriser plus de 70 lignes sous niveau 2, 16K.

Toutes ces modifications sont compatibles avec les versions disque et cassette décrites dans TRACE N° 6.

Il est toujours utile de faire 'RUN' avant d'attaquer une nouvelle page, pour 'vidanger' la zone du CLEAR encombrée par le texte précédent.
Pour terminer, je crois utile de rappeler que l'appellation 'traitement de texte' est très ambitieuse pour ce programme BASIC, à l'usage EXCLUSIF DES AMATEURS... Aucune comparaison n'est possible avec les logiciels du commerce qui comprennent toujours l'édition pleine page, avec le déplacement du curseur clignotant dans les quatre directions, les touches à répétition et surtout l'accès de nombreuses touches de contrôle assurant des fonctions très variées. Et ils sont naturellement toujours écrits en assembleur !

```
10 CLEAR 50 : CLEAR MEM/3*2 : DEFINT A,I-N : DEFSTR C,S : ME=FRE
(A$)/90 : DIM S(ME+3) :C=STRING$(133," ") : CS=C:REM28
50 IF N=JL-5 GOSUB 630:REM59
80 IF A=91 AND N<JL : N=N+1 : PRINT MID$(C,N,1); : GOTO 50:REMB4

150 IF A=26 AND POS(0)>7 PRINT S1; : N=N-8 : GOTO 50:REMD5
160 IF A=27 AND N<JL-8 AND POS(0)<55 PRINT S4; : N=N+8 : GOTO 50
:REM01
180 IF A=10 AND N>0 : N=N-1 : IF POS(0)>0 THEN PRINT CHR$(24); :
 GOTO 50 ELSE PRINT STRING$(63,25);CHR$(27); : GOTO 50:REMD8
400 M=M+1 : GOSUB 1200 : S(M)=LEFT$(C,N) : IF PEEK(16396)=201 TH
EN C=CS ELSE MID$(C,1)=CS:REM61
460 PRINT S(NO) : J=LEN(S(NO)) : PRINT S(NO);CHR$(29);STRING$(J/
64,27);CHR$(14); : M1=M : IF PEEK(16396)=201 THEN C=S(NO)+STRING
$(JL+1-J,32) : GOTO 40 ELSE MID$(C,1)=S(NO) : GOTO 40:REM09
1199 '----------------- alignement de la marge droite  4D
1200 J=N : I=1 : K=JL-N : IF K=0 OR K>4 RETURN:REM79
1210 I=I+1 : IF MID$(C,I,1)=C9 AND MID$(C,I-1,1)<>C9 THEN C=LEFT
$(C,I)+C9+MID$(C,I+1) : K=K-1 : I=I+1 : N=N+1:REM26
1220 IF K>0 AND I<JL+1 THEN 1210:REM47
1230 PRINT CHR$(29);STRING$(J/64,27);LEFT$(C,N); : RETURN:REM18
```

Roger Brousmiche

Super DIR sur modèle 2

Nous allons d'abord indiquer les services rendus par ce programme. Nous parlerons ensuite de quelques détails techniques.

COMPORTEMENT DU PROGRAMME

1) Demande du Numéro du Drive concerné (question indispensable pour les configurations multi-drives).
2) Demande du numéro de disquette (question utile).
3) Affichage des mots "Chargement Directory" à seule fin de faire patienter l'opérateur pendant quelques brèves secondes.
4) Affichage de la DIRECTORY "utilisateur" et d'un menu de commandes.

Le tableau peut contenir 96 noms, nombre maximum prévu pour les fichiers "non-système".
Sur la 1ère ligne figurent le Nom du Fichier N° 1, la date, le Numéro de DIRectory (DRIVE) et celui de la DISQuette.
On voit qu'il est facile de tirer de temps à autre une copier papier de ce tableau (par exemple lors des "BACKUPS").

Ce tableau est protégé de l'effet de rouleau.
En dessous, un menu propose les commandes suivantes :

L LOAD des Programmes BASIC

R RUN " " "

K KILL d'un Fich. quelconque (du tableau)

P PURGE (analogue à cde PURGE Système)

N NOMMER (reNommer) un fichier (du tableau))

D Détail d'un fichier

Avant de parler des commandes B et S précisons que les renseignements fournis par la commande D sont les suivants :

Type, Long.enr., LOF, Extensions, Sect.Alloués, Sect. Utilisés.

Les 2 dernières commandes sont les suivantes :

B BASIC (retour au, le pgm restant en mémoire)

S SYSTEM (retour au)

On met en oeuvre les commandes en frappant simplement la lettre correspondante. Pour les autres commandes, le programme demande ensuite le N° de fichier.

Pour les commandes L, R, et D, l'entrée du numéro de fichier déclenche l'exécution de la commande.

Pour les commandes K, P, et N, le nom du fichier est alors affiché et confirmation est demandée sous la forme classique (Y/N/Q).
Il faut noter que pour les commandes intervenant sur DISQUE le nom de fichier réellement utilisé est de la forme :

TOTO/BAS:Ø (ou :1, :2, :3)

Lorsqu'un fichier est tué, ou que son nom est changé, la case correspondante est immédiatement mise à jour en vidéo inversée.

Certaines commandes "restent" dans le programme (K, P, N, D), alors que les suivantes le quittent :

L, R, B, S

Immédiatement avant l'exécution de l'une ou l'autre de ces dernières commandes, un sous-programme de recopie d'écran en mémoire est appelé. Ceci permet en exécutant un autre petit programme d'avoir ultérieurement et très rapidement l'image exacte de la situation finale du tableau.

Il faut noter que les commandes L et R ne pouvant s' appliquer qu'aux seuls programmes BASIC, le nom de ceux-ci doit être muni d'une extension commençant par B (comme dant TOTO/BAS). Dans le cas contraire, l'ordre est rejeté. Si l'on est certain qu'il s'agit bien d'un programme BASIC, on aura toujours la ressource de renommer le fichier avant d'exécuter une commande L ou R.

Ceci nous amène à dire un mot sur les contrôles effectués (ou non effectués) par le DOS sur le nouveau nom :
Ainsi :

```
1TOTO/BAS      sera REJETE
 toto/basic     "     "    (ext. de plus de 3 car.)
TOTOSUPER/b     "     "    (nom trop long)
```

Mais :

```
TOTO%SUP        "  ACCEPTE (sous la forme TOTO)
TOTO/           "     "              "
TOTO?BAS:2      "     "              "
etc...
```

Il en résulte le risque suivant :

Nous sommes par exemple sur le DIRVE 2.
Nous changeons le nom TOTO/B en TOTO?BAS.
Le nom TOTO/B:2 est devenu TOTO:2 (au niveau du système).
Si maintenant nous voulons tuer TOTO?BAS, le programme va lancer l'ordre KILL "TOTO?BAS:2.
En raison de la présence intempestive du "?", le système interprétera l'ordre comme KILL TOTO (drive non défini) et tuera le premier TOTO venu.
C'est pour éliminer ce risque que dans la subroutine NAME, le programme complète les contrôles faits par le DOS en interdisant la présence dans le nom, d'un signe ou d'un espace, si ce n'est "/" en un seul exemplaire et à condition qu'il ne soit pas en dernière position.
Nous utilisons dans ce programme la routine SVC RENA-

ME, mais il faut noter que les choses seraient les mêmes avec l'ordre BASIC :

NAME "TOTO/B" AS "TOTO?BAS"

Seule la commande RENAME et la commande KILL sous DOS mentionnent le nom réellement pris en compte par le système. Ces commandes peuvent être rédigées par PEEK et POKE, mais la programmation est plus lourde et l'exécution plus lente.

CONSIDERATIONS TECHNIQUES

Le programme utilise quatre routines SVC (SCROLL, VIDRAM, RENAME, SORT et RAMDIR).
Nous ne ferons de commentaires que sur les deux dernières.

Routine SVC SORT

a) Cette routine utilise BUBBLE qui est un très mauvais algorithme (ce tri est battu par un tri QUICK-SORT en BASIC à partir de 700 éléments environ).

b) Ce tri ne peut trier que des éléments parfaitement contigus (c'est un tri direct).

c) Ce tri est utilisé ici car le nombre d'éléments à trier reste toujours faible et une routine SVC ne prend pas de place en mémoire BASIC.

Routine SVC RAMDIR

Le problème tient au fait qu'il faut indiquer a RAMDIR l'adresse à partir de laquelle la Directory doit être écrite en mémoire.
On procède de la façon suivante :

a) Dans le sous-programme d'initialisation une table de 96 éléments (éléments de 34 octets) est créee dans la zone chaînes de caractères.

b) Dans le sous-programme RAMDIR, on calcule l'adresse de Fin de cette table. C'est cette adresse de Fin (en réalité Fin - 1) qui sera le début de l'écriture de RAMDIR. Le (-1) est destiné à faire commencer les enregistrements par le 1er caractère du nom à la place du caractère ":".

CONCLUSION

Le présent programme est un outil puissant qui pallie certains défauts de la Directory du DOS. Il permet notamment de trouver rapidement un fichier, d'obtenir des copies sur papier qui soient utilisables, enfin de nettoyer au jour le jour les disquettes des fichiers d'essai, programmes abandonnés, etc...

Notons qu'un tel programme BASIC nous paraît impossible à réaliser sans la fonction PEEK. Comme la fonction NAME peut être réalisée par la routine SVC RENAME, il nous semble que PEEK et POKE remplacent avantageusement (à condition de ne pas utiliser POKE inconsidérement) NAME et OCT$.*

* (voir a ce sujet TRACE n° 5)

```
            Le nom ESPACE/B a été changé en BLANCS/B - Sortie par R45

ADRESSE/B...= 1|---(Lun-18-Jan-1982)---( DIR = 0 )---[GR2 ]------.09.10.00.----|
ANALMAF/B...= 2|EXDATM64....=21|INV999/DON..=40|SELEC2/B....=59|~...........=78|
BD2.........= 3|FAST/B......=22|MKDCVD/B....=41|SOCRESTE/B..=60|~...........=79|
BINAIR/B....= 4|FAST/BR.....=23|MORSO/B.....=42|ST050580....=61|~...........=80|
BITMAP/B....= 5|FAST1/B.....=24|MP/B........=43|SWAPDEC/B...=62|~...........=81|
BORNEG/B....= 6|FAST3/B.....=25|NBR/BAS.....=44|SWAPDEC1/B..=63|~...........=82|
BTD/B.......= 7|FAST5/B.....=26|P/B.........=45|SWAPDEC2/B..=64|~...........=83|
BUFFER/B....= 8|FASTRAND/B..=27|PATSITU/BAS.=46|TERSER/B....=65|~...........=84|
CADENAS/B...= 9|FASTRAND/BR.=28|PHONE/DON...=47|TID/B.......=66|~...........=85|
CADNUM/B....=10|FIELD/B.....=29|PHONE/DX1...=48|TIDVIR/B....=67|~...........=86|
CAR3/B......=11|FORNEXT/B...=30|PHONE.......=49|TRAV4/DON...=68|~...........=87|
CAR4/B......=12|FUSCBA/B....=31|PLUS/B......=50|TRI999/DON..=69|~...........=88|
CHR/B.......=13|FUSION/B....=32|PREMIERS/B..=51|TRIDOS/B....=70|~...........=89|
COMPACT/B...=14|IFTHEN/B....=33|PRINT/B.....=52|VIRGULE/B...=71|~...........=90|
COMSUB64....=15|IMPEC/B.....=34|QUICK25/B...=53|~...........=72|~...........=91|
DATM64......=16|INPUT/B.....=35|RALFA8/B....=54|~...........=73|~...........=92|
DICHOVIR/B..=17|INSERCAR/B..=36|RCDG/B......=55|~...........=74|~...........=93|
DOFILE......=18|INSERCH/B...=37|RE35/DON....=56|~...........=75|~...........=94|
DTB/B.......=19|INSERG/B....=38|RND/B.......=57|~...........=76|~...........=95|
BLANCS/B....=20|INT3/B......=39|ROTO/B......=58|~...........=77|~...........=96|
NOUVEAU Nom ? BLANCS/B
L,R,K,P,N,D ou B,S R
NR? 45
```

```
100 '                    Auteur: Bernard BESSE
200 '
300 '
400 CLEAR 10000
500 '
600 GOSUB 22100'-- INIT
700 GOSUB 1300'-- CHARGEMENT
800 CLS
```

```
900 GOSUB 6600'-- AFFICH
1000 GOSUB 7800'-- CMD
1100 GOSUB 19100'-- FIN
1200 '
1300 :REM< CHARGEMENT >
1400 PRINT"Chargement Directory"
1500 GOSUB 2300'-- RAMDIR
1600 IF NF<>0 THEN 1800
1700 PRINT VI$;"Directory VIDE";VN$:GOSUB 19100'-- FIN
1800 :IF NF=1 THEN 2000
1900 GOSUB 4800'-- SORT
2000 :GOSUB 5400'-- INVERSE
2100 RETURN
2200 '
2300 :REM< RAMDIR >
2400 '      Adr deb et fin V1$()
2500 XX=VARPTR(V1$(1)) :GOSUB 19400:AD=XX'-- ADR
2600 XX=VARPTR(V1$(MD)):GOSUB 19400:AF=XX'-- ADR
2700 AA=AD-AF: RR=(MD-1)*LG
2800 IFAA=RR THEN 3100
2900 PRINTVI$;"Adr.DEB & FIN non comp. av. TAILLE Table";VN$:GOSUB 19100'-FIN
3000 '      Ramdir
3100 :A1(7)=AF-1'-- l'adr pour ecrire la DIR est celle du der V1$ -1
3200 DEFUSR1=VARPTR(A1(1))
3300 A1(0)=USR1(0)
3400 IF A1(0)=0 THEN 3900
3500 PRINT"Erreur RAMDIR - DOS NR";A1(0)
3600 IF A1(0)=8 THEN PRINT"DRIVE N'EST PAS PRET"
3700 GOSUB 19100' -FIN
3800 '      Nbre de Fichiers/DIR: NF
3900 :NF=0
4000 FOR K= A1(7) TO A1(7)+3264 STEP 34
4100 CZ=PEEK(K):IF CZ=35 THEN 4500
4200 IFCZ<>58 THEN PRINT"Erreur marques enreg DIR al'adr=";K:GOSUB 19100'--FIN
4300 NF=NF+1
4400 NEXT K
4500 :A2(5)=K-33:A2(3)=AF'-- Parametres de SORT
4600 RETURN
4700 '
4800 :REM< SORT >
4900 DEFUSR2=VARPTR(A2(1))
5000 A2(0)=USR2(0)
5100 IFA2(0)>0THEN PRINT"Erreur Nr";A2(0);VI$;ER$(A2(0));VN$:GOSUB 19100'-FIN
5200 RETURN
5300 '
5400 :REM< INVERSE >
5500 ZF=MD\2:IF NF<ZF THEN ZF=NF
5600 FOR K=1 TO ZF
5700 SWAP V1$(K),V1$(MD+1-K)
5800 NEXT K
5900 '      Extraction Noms
6000 FOR K=1 TO MD
6100 IF K > NF THEN V2$(K)="~":GOTO 6300
6200 XP=INSTR(V1$(K),":")-1:V2$(K)=LEFT$(V1$(K),XP)
6300 :NEXT K
6400 RETURN
6500 '
6600 :REM< AFFICH >
6700 DEFUSR3=VARPTR(A3(1))
6800 A3(0)=USR3(H)
6900 PRINT @(0,0),V2$(1);STRING$(12-LEN(V2$(1)),".");"= 1|---(";JS$;"-";QU$;'
MO$;"-";AN$;")---( DIR =";DR;")---[ ";DK$;" ]";STRING$(20,"-");"|";
7000 FOR LI=2 TO NL+1
7100 FOR K=LI TO MD STEP NL
7200 PT=12-LEN(V2$(K)):T=((K-2)\NL)*16:PRINTTAB(T)V2$(K);STRING$(PT,".");"="
INTUSING"##";K;:PRINT"|";
7300 NEXT K
7400 NEXT LI
7500 PRINT
7600 RETURN
7700 '
7800 :REM< CMD >
```

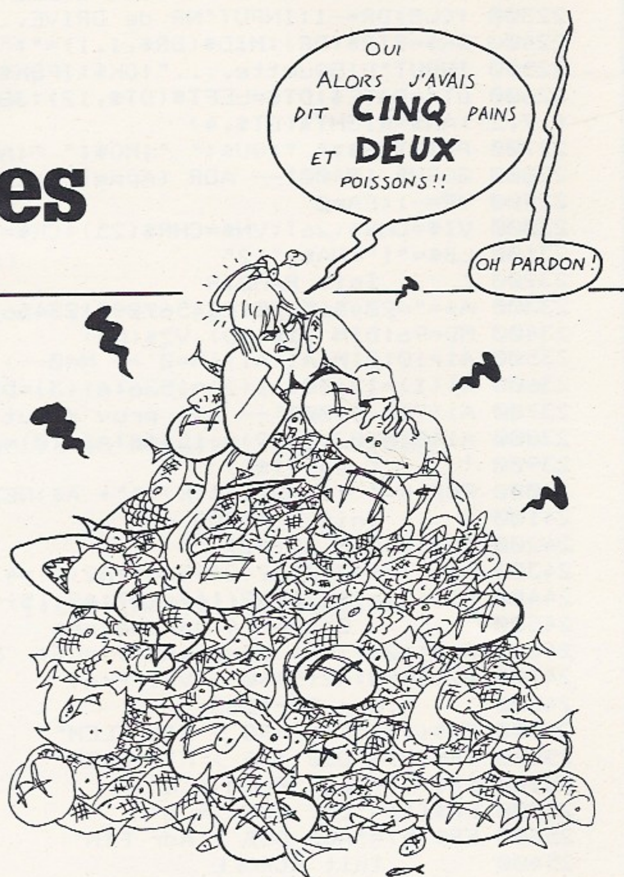
```
7900 :GOSUB 17300'-- SAISIE
8000 ON CD GOSUB 8300,8300,9400,10300,14100,15300,16700,17000
8100 GOTO 7900
8200 '
8300 :REM< LOAD/RUN >
8400 XP=INSTR(V2$(NR),"/B"):IFXP<>0 THEN GOSUB 9100:GOTO 8600'--IMAGE
8500 PRINT"Load/Run de ";V2$(NR);" interdit (/B)":GOTO 8900
8600 :CLS:IF CD$="L" THEN PRINT"LOAD";ELSE PRINT"RUN";
8700 GOSUB 13100'--NOM ACTUEL
8800 PRINT" DE  :    ";W1$;"DEMANDE":IFCD$="L"THEN LOAD W1$ ELSE RUN W1$
8900 :RETURN
9000 '
9100 :REM< IMAGE >
9200 DEFUSR1=VARPTR(A4(1)):A4(0)=USR1(0):RETURN
9300 '
9400 :REM< KILL >
9500 :PRINT"KILL ->";VI$;V2$(NR);VN$;"(Y/N/Q)";:YN$=INPUT$(1):PRINTYN$
9600 IF YN$<>"Y" THEN 10000
9700 GOSUB 13100'--NOM ACTUEL
9800 KILL W1$:V2$(NR)="~":XC=POS(0):YC=ROW(0):GOSUB 13400'--PEINTURE
9900 GOTO 10100
10000 :IF YN$<>"N" AND YN$<>"Q"THEN 9500
10100 :RETURN
10200 '
10300 :REM< NAME >
10400 :PRINT"C'est ";VI$;V2$(NR);VN$;" (Y/N)";:YN$=INPUT$(1):PRINTYN$
10500 IF YN$="N" THEN12900
10600 IF YN$<>"Y" THEN 10400
10700 :NV$="":INPUT"NOUVEAU Nom ";NV$:IF NV$=""THEN 12900
10800 BF=0:LV=LEN(NV$)
10900 FOR K=1 TO LV
11000 LSET CK$=MID$(NV$,K,1)
11100 IFCK$<>"/"THEN 11500
11200 IF K=LV THEN 11800
11300 BF=BF+1:IF BF>1 THEN 11800
11400 GOTO 11900
11500 :IFCK$>="A"ANDCK$<="Z"THEN 11900
11600 IFCK$>="0"ANDCK$<="9"THEN 11900
11700 IFCK$>="a"ANDCK$<="z"THEN 11900
11800 :PRINT"Espace(s) ou signes(s) a supprimer":GOTO 10700
11900 :NEXT K
12000 GOSUB 13100:LSET W2$=NV$'-- NOM ACTUEL
12100 XX=VARPTR(W1$):GOSUB 19400:A5(3)=XX'-- ADR
12200 XX=VARPTR(W2$):GOSUB 19400:A5(5)=XX'-- ADR
12300 DEFUSR5=VARPTR(A5(1)):A5(0)=USR5(0)
12400 IFA5(0)=0THEN 12800
12500 IFA5(0)=11THEN PRINT"CE NOM EXISTE DEJA":GOTO 10700
12600 IFA5(0)=19THEN PRINT"FAUTE(S) DANS NOM":GOTO 10700
12700 PRINT"Origine RENAME - ERREUR DOS";A5(0):GOTO12900
12800 :V2$(NR)=NV$:XC=POS(0):YC=ROW(0):GOSUB 13400:GOTO 12900'--PEINTURE
12900 :RETURN
13000 '
13100 :REM< NOM ACTUEL >
13200 LSET W1$=V2$(NR):XP=INSTR(W1$," "):MID$(W1$,XP)=DR$:RETURN
13300 '
13400 :REM< PEINTURE >
13500 NK=NR-1:IF NK=0 THEN Y=0:X=0:GOTO 13800
13600 Y=(NK-1)MOD NL +1
13700 X=((NK-1)\NL)*16
13800 :L=LEN(V2$(NR)):PRINT@(Y,X),VI$;V2$(NR);STRING$(12-L,".");VN$:PRINT@(YC,XC
),;
13900 RETURN
14000 '
14100 :REM< PURGE >
14200 FOR NR=1 TO MD
14300 IF V2$(NR)="~"THEN 15000
14400 :PRINT"Kill Nr";VI$;NR;VN$;"(Y/N/Q)";:YN$=INPUT$(1):PRINTYN$
14500 IF YN$="N"THEN 15000
14600 IF YN$="Q"THEN15100
14700 IF YN$<>"Y"THEN 14400
14800 GOSUB 9400'-- KILL
14900 IF YN$="Q" THEN 15100
```

```
15000 :NEXT NR
15100 :RETURN
15200 '
15300 :REM< DETAIL >
15400 LSETTF$=MID$(V1$(NR),16,1)
15500 LSETRL$=MID$(V1$(NR),17,1):RL=ASC(RL$)
15600 IFTF$="F"AND RL=0 THEN RL=256
15700 LSETXT$=MID$(V1$(NR),18,1):XT=ASC(XT$)
15800 LSETNS$=MID$(V1$(NR),21,2):NS=CVI(NS$)
15900 LSETLB$=MID$(V1$(NR),23,1):LB=ASC(LB$)
16000 LF!=(NS-1)*256+LB+RL:LF!=INT(LF!/RL)
16100 LSETNA$=MID$(V1$(NR),19,2):NA=CVI(NA$)
16200 PRINTVI$;V2$(NR);VN$;
16300 PRINT" # Type=";TF$;" | LRL=";RL;"| LOF=";LF!;"| Ext=";XT;"|";
16400 PRINT" Sa=";NA;"| Sut=";NS;"#"
16500 RETURN
16600 '
16700 :REM< BASIC >
16800 GOSUB 9100:CLS:CLOSE:PRINT"PROG. toujours en Memoire":END'--IMAGE
16900 '
17000 :REM< SYSTEM >
17100 GOSUB 9100:CLS:CLOSE:SYSTEM'--IMAGE
17200 '
17300 :REM< SAISIE >
17400 :PRINTVI$;"L,R,K,P,N,D ou B,S";VN$;" ";:CD$=INPUT$(1):PRINTCD$
17500 IF CD$="L" THEN CD=1:GOTO 18400
17600 IF CD$="R" THEN CD=2:GOTO 18400
17700 IF CD$="K" THEN CD=3:GOTO 18400
17800 IF CD$="N" THEN CD=4:GOTO 18400
17900 IF CD$="P" THEN CD=5:GOTO 18900
18000 IF CD$="D" THEN CD=6:GOTO 18400
18100 IF CD$="B" THEN CD=7:GOTO 18900
18200 IF CD$="S" THEN CD=8:GOTO 18900
18300 GOTO 17400
18400 :PRINTVI$;"NR";VN$;:N$="":INPUT N$
18500 IF N$="" THEN 17400
18600 NR=INT(VAL(N$))
18700 IF NR<=0 OR NR>MD THEN 18400
18800 IF V2$(NR)="~" THEN PRINT "~ n'existe pas":GOTO 18400
18900 :RETURN
19000 '
19100 :REM< FIN >
19200 PRINT"FIN Anormale":CLOSE:END
19300 '
19400 :REM< ADR >
19500 LX=1:GOSUB 19800:X1=XX:GOSUB 19800:X2=XX'-- DEPLACEMENT
19600 X1=PEEK(X1):X2=PEEK(X2):XX!=X1+256*X2:IFXX!<32768 THEN XX=XX!:RETURN ELSE
XX=XX!-65536:RETURN
19700 '
19800 :REM< DEPLACEMENT >
19900 XX=XX+LX
20000 :RETURN
20100 '
20200 :REM< ERR >
20300 IF ERR<>6 THEN 20600
20400 IF ERL <> 19900THEN 21500
20500 GOSUB 21700: RESUME 20000'--OVERFLOW
20600 :IF ERR <> 53 THEN 21200
20700 PRINT@(23,27),VI$;"*** ";V2$(NR);" NON TROUVE ***";VN$
20800 IF ERL<>8800THEN 21000
20900 RESUME 6600
21000 :IF ERL<>9800THEN 21500
21100 RESUME 10100
21200 :IF ERR<>54 THEN 21500
21300 CLOSE
21400 PRINT@(23,27),ERR;VI$;"*** ";V2$(NR);" non PGM basic ***";VN$:RESUME 6600
21500 :ON ERROR GOTO 0
21600 '
21700 :REM< OVERFLOW >
21800 IF XX>0THEN XX=XX-32768+LX-32768 ELSE XX=XX+32768+LX+32768
21900 RETURN
22000 '
```

```
22100 :REM< INIT >
22200 DEFINTA-Z:ON ERROR GOTO 20200
22300 :CLS:DR=-1:INPUT"NR de DRIVE..";DR:IF DR <0 OR DR >3 THEN 22300
22400 DR$=STR$(DR):MID$(DR$,1,1)=":"
22500 INPUT"DISQuette....";DK$:IFDK$=""THEN22300 ELSE DK$=LEFT$((DK$+"     "),4)
22600 DT$=DATE$:DT$=LEFT$(DT$,12):JS$=LEFT$(DT$,3):MO$=MID$(DT$,4,3):QU$=MID$(DT
$,7,2):AN$=RIGHT$(DT$,4)
22700 PRINTJS$;" ";QU$;" ";MO$;" ";AN$
22800 GOSUB 19400'-- ADR (Appel Factice pour creer var. de la Routine)
22900 VR=-1:FA=0
23000 VI$=CHR$(26):VN$=CHR$(25):CK$="1":TF$="1":RL$="1":XT$="1":NS$="12"
23100 LB$="1":NA$="12"
23200 '     Init RAMDIR
23300 A$="~234567890123456789012345678901234"
23400 MD=96:DIM V1$(MD),V2$(MD)
23500 A1=10:DIM A1(A1):N=0'-- N=0--> tte la DIR/ DR nr Dir
23600 A1(1)=13630:A1(2)=1536:A1(3)=DR:A1(4)=3584:A1(5)=N:A1(6)=8448
23700 A1(7)=-11000'-- val prov debut RAMDIR
23800 A1(8)=207:A1(9)=-15616:A1(10)=15099
23900 '     Crea V1$()
24000 FOR K=1 TO MD:V1$(K)=""+ A$:NEXT K
24100 '     Init SORTDOS
24200 A2=16:DIM A2(A2)
24300 A2(1)=14398:A2(2)=8669:A2(4)=4352:A2(6)=1536:A2(8)=3584:A2(10)=9728
24400 A2(12)=11776:A2(14)=207:A2(15)=-15616:A2(16)=15099
24500 '      Entree 4 parametres
24600 P=0:A2(7)=P: LG=34:A2(9)=LG:  SW=0:A2(11)=SW:  LK=12:A2(13)=LK
24700 A2(3)=0:A2(5)=0'--Val Prov
24800 '     Err SortDOS
24900 ER$(1)="FIN ARG > FIN ELEM"
25000 ER$(2)="DEB ARG > FIN ELEM"
25100 ER$(3)="Lg. ELEM = 0"
25200 ER$(4)="Lg. ARG  = 0"
25300 ER$(5)="Adr DEB > Adr FIN"
25400 '     Init SCROLL
25500 A3=4:DIM A3(A3):A3(1)=766:A3(2)=18112:A3(3)=6974:A3(4)=-13873
25600 '     Init. AFFICH
25700 CL=5:NL=(MD-2)\CL +1:H=NL+1
25800 IFH<0THEN H=0 ELSE IF H>22 THEN H=22
25900 '               Init IMAGE
26000 A4=6:DIM A4(A4):A4(1)=24126:A4(2)=1536:A4(3)=1:A4(4)=8448:A4(5)=-4090:A4(6
)=-13873
26100 '               Init RENAME
26200 W1$=STRING$(15," "):W2$=STRING$(15," ")
26300 A5=8:DIMA5(A5):A5(1)=12094:A5(2)=8448:A5(3)=0:A5(4)=4352:A5(5)=0
26400 A5(6)=207:A5(7)=-15616:A5(8)=15099
26500 RETURN
26600 '******
```

Bernard Besse

La magie des nombres

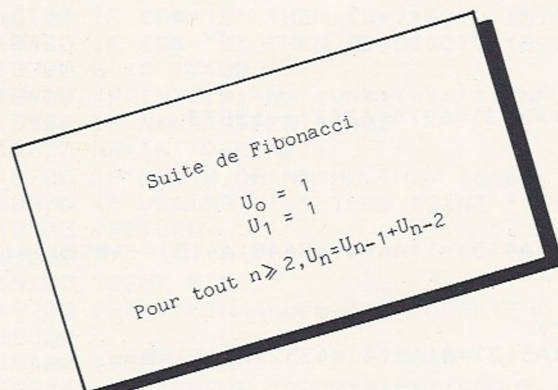


"LA MAGIE DES NOMBRES" OU
"L'INFORMATIQUE AU SERVICE DES MATHEMATIQUES"

Léonardo Fibonacci (dit Léonard de Pise) inventait en 1202 la suite dite Fibonacci. Or, cette suite possède une fantastique propriété :

$$\lim_{n \to +\infty} \frac{U_{n+1}}{U_n} = \text{Constante}$$

Cette constante qui est l'une des deux racines de l'équation du second degré $x^2-x-1=0$, fut plus tard appelée "nombre d'or". Autrement dit, le carré du nombre d'or est égal au nombre d'or plus 1. Ce nombre fut par la suite énormément utilisé en peinture pour déterminer les proportions d'un tableau.



Une chose est évidente : la suite de Fibonacci s'adapte très bien à la programmation. Voici donc un programme qui donne pour tout rang n, sa valeur de Fibonacci ainsi que celle du nombre d'or associé.

```
10:INPUT "N (>2)
   ?";N:U0=1:U1=1
11:FOR I=2TO N:U=
   U0+U1:PRINT I;
   U;U/U1
12:U0=U1:U1=U:
   NEXT I
```

Maintenant intervient la déduction informatisée : d'après l'équation $x^2-x-1=0$, le nombre d'or $\partial = \frac{1+\sqrt{5}}{2}$ l'autre valeur étant une "solution bâtarde"). Le programme qui suit calcule, pour tout rang n, la valeur du rapport de ∂^n sur U_n :

```
10:INPUT "N (>2)
   ?";N:U0=1:U1=1
11:FOR I=2TO N:U=
   U0+U1
12:U0=U1:U1=U:
   NEXT I
13:PRINT N;(((1+√
   5)/2)^N)/U1
```

Avec ce programme, on remarque que plus n est grand, plus le rapport tend vers une valeur limite à :

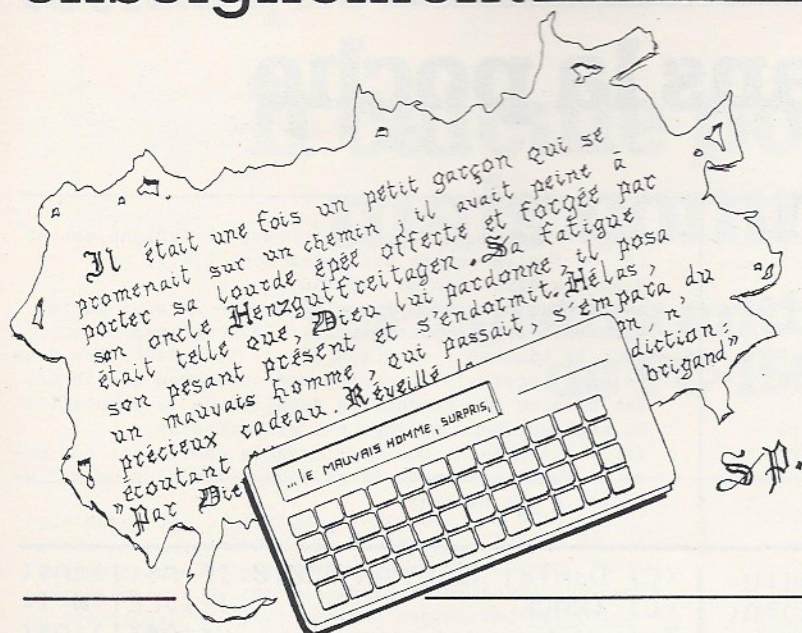
$\simeq 1,381966011$

Or, tout le monde (!) aura remarqué que cette valeur est égale à $3-\partial$; ce qui nous conduit à une formule asymptotique :

$U_n \simeq \frac{\partial n}{3-\partial}$ ... (vérifiez, ça marche !)

On remarquera que si $n \geqslant 10$ la marge d'erreur est inférieur à 0,01 % !
Et voilà, par l'informatique, une constatation a permis l'élaboration d'une formule mathématique...

Daniel Glazman



Il vous raconte des histoires

Le PC-1211 ou TRS80 PC1 relié à un magnétophone à cassette est un ordinateur parlant (et chantant). Voici un programme qui s'adresse à tous les parents, français ou étrangers, qui ont un enfant en âge d'apprendre à lire et qui peut ainsi progresser en lecture en se divertissant, comme avec un "livre-disque". Une "histoire" se déroule par séquences de 43 phrases au maximum, avec les phrases enregistrées sur cassette. L'exécution débute en mode DEF avec SPC. L'ordinateur signale alors qu'il faut mettre un numéro en ligne 2 du programme (numéro ou étiquette de 1 ou plusieurs caractères dont seuls les 7 premiers sont utiles) qui indique la position de la première séquence de l'histoire sur la cassette. Après cela, faire SHFT A. Toutes les opérations a effectuer sont précisées. Quand l'ordinateur demande un nombre de lignes, entrer un petit nombre pour un débutant en lecture, ou taper sur ENTER : il saura que c'est 43. Dans une histoire, la 2ème fois que l'ordinateur affiche CASSETTE PRETE ? il faut supprimer l'étiquette de la ligne 2 sans laisser d'espace (INPUT"") puis refaire SHFT A. Ce sera ensuite automatique pour les séquences suivantes. Mais si vous avez enregistré chaque séquence d'une même histoire avec la même étiquette, cela évitera cette manipulation mais rendra le retour en arrière plus aléatoire.

Pour enregistrer les lignes écrites, faire SHFT =. Entrer 7 caractères à la fois, espaces, virgules, traits d'union ou apostrophe compris. Une ligne affichera 3 groupes de 7 caractères ensemble.
A l'enregistrement des phrases parlées, on pourra modifier la durée du son en changeant la valeur 124 en lignes 9 et 14. 124 fait durer un peu moins de 10 s suivant l'appareil utilisé. Une baisse de 1 unité augmente le temps de 0,59 s en-dessous de 120. On entend un sifflement de 5,2 s puis des séries de grésillements

A l'enregistrement, il vaut mieux ne pas parler pendant les première et dernière secondes. A l'audition, on n'entendra guère les sifflements et grésillements. Après une ligne enregistrée, on entend un "bip". Après l'enregistrement d'une séquence écrite, on peut s'arrêter et éteindre l'ordinateur. On enregistre le son avec SHFT B. SHFT C fait démarrer la lecture.

```
*--- TEXTE ---*
   -=-=-

ELIE RANCHOUX

  2:INPUT #"?":
    PRINT " BAIS
    SE LE SON.":
    PRINT " OTE
    LA FICHE GRI
    SE.":RETURN
  3:PRINT " ";C$
    ;D$;E$:
    RETURN
  4:" "PAUSE " *
    --- TEXTE A
    LIRE ---*":
    PAUSE " CHOI
    X DE LA SERI
    E :":PAUSE "
     MODE PRO :
  5:PRINT "L.2 :
     NO ? ,PUIS
    SHFT A
  6:"A"PRINT " C
    ASSETTE PRET
    E ?":PRINT "
     FICHE GRISE
     ?":PRINT "S
    ON: + , STAR
    T":GOSUB 2
  7:B=43:INPUT "
    NOMBRE : ";B
  8:FOR A=3TO 3B
    STEP 3:C$=A$
    (A),D$=A$(A+
    1),E$=A$(A+2
    ):GOSUB 3
 -9:PRINT #"";A(
    124):GOSUB 3
    :NEXT A:GOTO
    6
 10:"="CLEAR :
    USING "####"
    :FOR A=1TO 1
    29:PAUSE "EC
    RIS LE";A;"E
     GROUPE,":
    INPUT A$(A+2
    ):GOTO 15
 11:PRINT "NO EN
     L.13; B
 12:"B"PAUSE "EN
    REGISTREMENT
    ":PRINT "MIC
    , REMOTE; RE
    C, SON
 13:PRINT #"?":
    PRINT "MICRO
     ON EN MIC;
    REC
 14:FOR B=4TO 13
    0STEP 3:A=B+
    1:PAUSE "LIR
    E:":PAUSE A$
    (B-1);A$(B);
    A$(A):PRINT
    #"";A(124):
    NEXT B
 15:IF A/3=INT (
    A/3BEEP 1
 16:NEXT A:GOTO
    11
 17:"C"PRINT "":
    GOTO 4
```

Elie Ranchoux

POQUETTE

utilitaire

Le tri dans la poche

Matériel : PC-1500

Qui n'a jamais fait un programme de gestion de carnet téléphonique, d'annuaire, etc...? Et qui n'a pas eu de problèmes pour y inclure une routine de tri alphabétique ?

Voici la liste de deux routines, l'une pour un tri numérique, l'autre alphanumérique. Dans la première, le tri s'effectue sur le tableau A(*) et utilise les variables U, W et I.

Dans le cas du tri alphanumérique, le tableau est A$ (*) et les variables utilisées sont U,W,I.
N est la dimension du tableau.
La méthode de tri employée n'est pas la plus rapide mais elle est la plus simple : le programme scrute tout le tableau ; si $A(x) < A(x+1)$ il échange les deux valeurs ($0 \leqslant x \leqslant N-1$). Le programme considère que le tri est termine si, pendant la totalité de la scrutation du tableau aucun échange n'a été effectué.
Voilà ! A présent que vous avez la routine et son mode de fonctionnement, il ne vous reste plus qu'à l'adapter à vos besoins...

```
(C) Daniel GLAZMAN
(C) TRACE
Tri numerique

65000:W=-1:FOR I=1
      TO N
65010:IF A(I)<A(I-
      1)LET W=1:U=
      A(I):A(I)=A(
      I-1):A(I-1)=
      U:BEEP 1,0
65020:NEXT I:IF W<
      >-1GOTO 6500
      0
65030:RETURN
```

```
(C) Daniel GLAZMAN
(C) TRACE
Tri alphanumerique

65000:W=-1:FOR I=1
      TO N
65010:IF A$(I)<A$(
      I-1)LET W=1:
      U$=A$(I):A$(
      I)=A$(I-1):A
      $(I-1)=U$:
      BEEP 1,0
65020:NEXT I:IF W<
      >-1GOTO 6500
      0
65030:RETURN
```

Daniel Glazman

nouveau matériel

Sharp PC-1401

Ca y est ! Un nouveau pas dans l'informatique de poche vient d'être franchi par Sharp ! En effet, le dernier-né de la famille Sharp "pocket" est peut-être le plus étonnant, jugez donc :

Sa taille est d'1,5 fois celle du PC-1251 mais sa puissance est de loin supérieure :
La machine est divisée en 2 parties : l'une des parties est l'ordinateur Basic, l'autre est la calculatrice scientifique.

Commençons par la partie Basic : elle est très ressemblante au PC-1251, sans touches programmables mais avec des touches préprogrammées (comme sur PC-1245).

Le Basic occupe (tenez-vous bien !) 40 Koctets en mémoire, la mémoire RAM étant de 4,2 K ; de plus le Basic et la machine font que le PC-1401 est compatible avec les PC-1211, PC-1212, PC-1251, PC-1245...
Du côté du langage : toutes les fonctions d'un Basic qui se respecte sont présentes (sauf toujours le ELSE ...), seulement voilà, les fonctions scientifiques ont été enrichies : des trigonométriques, hyperboliques, opérations en hexa, et toutes les fonctions statistiques existant sur la FX-702P de Casio.

Un autre point qui n'est pas des moindres : le PC-1401 a une précision de calcul qui semble excellente.

Mais alors me direz-vous, si la PC-1401 ressemble tellement au FX-702P du côté Basic, qu'est-ce qui fait son atout ? Eh bien c'est la partie calculatrice scientifique : en effet, la moitié droite du clavier de 76 touches est réservée à la calculatrice. Tous les calculs s'affichent sur l'écran à cristaux liquides de 16 caractères (contraste règlable). Il existe même un indicateur vous montrant que vous utilisez les parenthèses. Bien entendu, en mode "calc", vous disposez d'une mémoire spécialisée pour stocker un résultat intermédiaire. Pour repasser en Basic, il suffit d'appuyer sur une touche.

Mais regardez jusqu'où est poussé le raffinement : vous vous trouvez en mode Basic, vous appuyez sur une touche "calculatrice", la fonction correspondante s'affichera en toutes lettres...
Ex : Si vous appuyez sur ARC, HYP, SIN il s'affichera AHS (arc hyperbolic sine) !
Pas mal, non ?

Du côté extension, le PC-1401 sera connectable à une interface cassette/imprimante référencée CE-126P.
L'imprimante est thermique de 24 caractères.
Son prix ? Aux alentours de 1600 FF ttc.

Le PC-1401, dans son étui rigide, peut donc être considéré comme le premier d'une lignée de machines alliant un Basic performant à une très haute précision de calcul.

Daniel Glazman

Il calcule pour vous mais comment ?

Les registres arithmétiques du PC-1500

Pour effectuer les calculs arithmétiques utilisés notamment par le BASIC, le PC-1500 possède un jeu d'instructions spécialisées qui sont directement interprétées par le microprocesseur. Elles possèdent leurs propres codes et leurs propres registres. Attention : ne pas confondre avec les instructions et les registres du microprocesseur dont il est question dans l'autre article qui suit !...

Le PC-1500 possède 7 registres arithmétiques de 8 octets chacuns, situés à partir de &7A00, en RAM SYSTEM.

7A00	X
7A08	Z
7A10	Y
7A18	U
7A20	V
7A28	W
7A30	S

Un registre arithmétique peut contenir trois types d'information, selon la valeur de son 5ème octet que l'on appellera caractéristique. Ainsi un registre arithmétique pourra représenter :

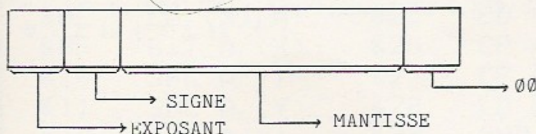
- . un nombre décimal
- . un entier 16 bits signé
- . une variable

On notera "C" la caractéristique et "RA" un registre arithmétique.

REPRESENTATION D'UN NOMBRE DECIMAL

Il faut pour cela que C < B2 (HEXA)
Le nombre est compris entre $-9.99999999\ 10^{99}$ et $9.99999999\ 10^{99}$

Le nombre se code comme suit :

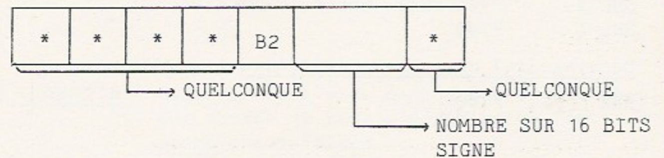


- . L'exposant est exprimé par un octet sur 8 bits signés.
- . Le signe de la mantisse par : 00 positif
  80 négatif
- . La mantisse est codée en DCB sur 5 octets.

REPRESENTATION D'UN NOMBRE ENTIER

Il faut pour cela que C = B2 (HEXA)
Le nombre est compris entre : -32768 et + 32767, donc sur 16 bits signés.

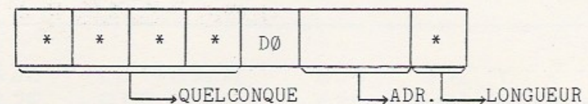
Le nombre se code comme suit :



REPRESENTATION D'UNE VARIABLE

Il faut pour cela que C > B2 (en général C = D0)

La variable se code comme suit :



ADR représente l'adresse de la variable.

- . Si la variable est alphabétique, le dernier octet représente la longueur de la chaîne.
- . Si la variable est numérique alors la longueur vaut 88 (HEXA).

OPERATIONS SUR LES REGISTRES ARITHMETIQUES

CHARGEMENTS

Chargements sur 8 octets

LDS,Y	F701	LDX,S	F737
LDS,X	F707	LDX,Y	F73D (56)
LDY,X	F70D (E6)	LDX,(BC)	F73F
LDY,S	F715 (68)	LDY,(BC)	F717
LDS,(BC)	F703	LD (DE),(BC)	F741
LD (DE),X	F711		

Chargements sur 7 octets (sans l'exposant) :

$\overline{\text{LD}}$X,Y	F71B	LDZ,X	F72F (78)
$\overline{\text{LD}}$X,Z	F71F	LDX,(BC)	F721
$\overline{\text{LD}}$Z,Y	F725	LDY,(BC)	F72B
$\overline{\text{LD}}$Y,X	F729 (82)	LD(DE),(BC)	F733

ECHANGES

```
EXC X,S     F7B5        EXC X,(DE)   F7BB
EXC X,Y     F7B9 (66)   EXC (BC),(DE)F7BD
```

MISES A ZERO

Zéro sur 8 octets :

```
CLX         F757 (EC)   CL(BC)       F75B
CLY         F753
```

Zéro sur 7 octets (sans l'exposant) :

```
LDX,Ø       F75F (76)   LDW,Ø        F74F
LDY,Ø       F753        LD(BC),Ø     F761
LDZ,Ø       F74B
```

DECALAGES

Décalage d'½ octet vers la droite :

(On code 2 chiffres décimaux par octet)

```
SHR X       F775 (74)   SHR Z        F76D
SHR Y       F771        SHR W        F769
```

Décalage d'½ octet vers la gauche :

```
SHL X       F79C (EA)   SHL W        F794
SHL Z       F798
```

Décalage d'1 octet vers la droite :

```
RRX         F780
```

OPERATIONS DIVERSES

```
ABS X       F6FB (6C)   ABS X,Y      F6E6 (7C)
INV X       F7EB        MU2          F81C
MU3         F820        LDR          F7BØ (54)
```

FONCTIONS SUR LES RA

OPERATEURS DIADIQUES

```
. X = X + Y : EFBA(FØ) pas de registres modifiés à part X
. X = X - Y : EFB6        "          "         "    "    "
. X = X * Y : FØ1A(7E) Y et S non modifiés
. X = X / Y : FØ84(58) Y, W et S non modifiés
. X = X ∧ Y : F89C     S = Y, sinon tout est modifié
. X = Y / X : FØ82     X,W et S non modifiés
```

OPERATEURS MONOADIQUES

```
. X = √X    : FØE9      U,V,W et S inchangés
. X = X²    : FØ19      Y = X et S non modifié
. X = 1/X   : FØ8Ø (6E) Y = X et W, S non modifiés
. X = LNX   : F161      W = Ø, S non modifié
. X = LOGX  : F165      "    "    "     "
. X = EXPX  : F1CB      "    "    "     "
. X = 1Ø↑X  : F1D4      Y = W = Ø, S non modifié
. X = SINX  : F3A2      tous les registres sont utilisés
. X = COSX  : F391        "     "     "          "
. X = TANX  : F39E      W = Ø, S non modifié
. X = ASNX  : F49A      W = Ø, S = X
. X = ACSX  : F492      W = Ø, S = X
. X = ATNX  : F496      Z = Ø, S non modifié
. X = DEGX  : F531      W non modifié
. X = DMSX  : F564      tous les registres sont modifiés
. X = ABSX  : F597      seul X est modifié
. X = SGNX  : F59D        "   "   "     "
. X = INTX  : F5BE      Y = FRCX, pas d'autres modification
. X = RNDX  : F5DD      tous les registres sont modifiés
```

- Les résultats qui se trouvent dans X sont normalisés.
- Si il y a un dépassement ou une erreur le bit carry est levé, et H = ERREUR.
- En ce qui concerne les registres du processeur, seuls les poids forts de BC et DE sont intacts.

  B = 7A
  C = 7A

- Pour normaliser un registre, utiliser la macro E8 : NORM.

COMPARAISONS DES REGISTRES ARITHMETIQUES

COMPARAISONS NUMERIQUES

Les deux nombres sont stockés dans les registres arithmétiques X et Y.
Le type de l'opérateur de comparaison est stocké dans l'accumulateur selon le tableau suivant.

	b7	b6	b5	b4	b3	b2	b1	b0
A=	Ø	Ø	Ø	Ø	=	>	<	<>

L'adresse de la routine est : DØD2.

Au retour, si le test est VRAI alors : X = 1 et le flag Z = Ø
si le test est FAUX alors : X = Ø et le flag Z = 1

Exemple : X≥Y ? :

```
LDX, 1er nombre
LDY, 2ème nombre
LDA,Ø6              [b3 + b2]
CP  [CALL DØD2]
```

COMPARAISONS ALPHABETIQUES

Les deux chaînes sont stockées dans les registres X et Y.
Le type de la comparaison est stocké dans l'accumulateur comme pour une comparaison numérique.
L'adresse de la routine est : DØF9.
Le type de retour est le même que pour les comparaisons numériques.

COMPARAISONS A ZERO

Cela revient à comparer le 2ème octet du registre quand celui-ci est formaté.

Exemple : CPX,Ø se traduit par : CP(7AØ2),Ø
et Z = 1 si X = Ø

Pascal Abrivard

Les instructions assembleur du PC-1500

A force de recherches et d'appuis sur la touche RESET, j'ai appris que :

. le LH 5801 (processeur du PC-1500) est un 8 bits capable d'accéder directement à 128 Ko (!) de mémoire.
. le cycle interne de la machine est de 1,3 MHz.
. les registres du MPU sont au nombre de 8 :

P	: 16 bits	Program pointer
S	: 16 bits	Stock pointer
W	: 16 bits	?
A	: 16 bits	Accumulateur
E (ou F?)	: 8 bits	Flags register
U,X,Y	: 3 registres de 16 bits	

Lorsque j'ai acheté l'imprimante CE-150, la première chose que j'ai faite a été de sortir <u>toute la ROM</u> sous forme d'octets hexa et caractères ASCII. A partir de là, il m'a été facile de trouver le code de RET (retour de sous-programme) et les codes d'accès immédiats aux registres.
Après, toutes les instructions furent trouvées au fur et à mesure.

QUELQUES ECLAIRCISSEMENTS SUR LES CODES :

. Le code &FD est un préfixe : tout code de la 2ème table doit être précédé de &FD
. LDD : Load & Decrement
. LDI : Load & Increment
. (n) = &FF00+n
. RCF : Reset Cary Flag
. SCF : Set Cary Flag
. DADC,DSBC : Operation en DCB
. AEX échange les 8 premiers bits avec les derniers bits (de A)
. SPU : SET U
. RPU : RESET U
. RVF : RESET V
. ENT n,d;EVL+d;ERR;RTS;ERR1   Affaire à suivre...
. RBF : Reset B

<u>ATTENTION</u> : Contrairement à de nombreux processeurs, une donnée 16 bits s'écrit poids fort <u>avant</u> poids faible.

Ex : LD A, (&7800)   A57800

Dans le cas d'une donnée 16 bits <u>et</u> d'une donnée 8 bits, la 16 bits s'écrit <u>avant</u> <u>la</u> 8 bits.

Ex : ADD (&7800),&84   EF780084

&00	SBC A,Xl	&17	CP A,(Y)	&2E	LD (U),A	&54	INC Y
&01	SBC A,(X)	&18	LD Yh,A	&2F	BIT A,(U)	&55	LDI A,(Y)
&02	ADC A,Xl	&19	AND A,(Y)	&38	NOP	&56	DEC Y
&03	ADC A,(X)	&1A	LD Yl,A	&40	INC Xl	&57	LDD A,(Y)
&04	LD A,Xl	&1B	OR A,(Y)	&41	LDI (X),A	&58	LD Yh,n
&05	LD A,(X)	&1C	DSBC A,(Y)	&42	DEC Xl	&59	AND (Y),n
&06	CP A,Xl	&1D	XOR A,(Y)	&43	LDD (X),A	&5A	LD Yl,n
&07	CP A,(X)	&1E	LD (Y),A	&44	INC X	&5B	OR (Y),n
&08	LD Xh,A	&1F	BIT A,(Y)	&45	LDI A,(X)	&5C	CP Yh,n
&09	AND A,(X)	&20	SBC A,Ul	&46	DEC X	&5D	BIT (Y),n
&0A	LD Xl,A	&21	SBC A,(U)	&47	LDD A,(X)	&5E	CP Yl,n
&0B	OR A,(X)	&22	ADC A,Ul	&48	LD Xh,n	&5F	ADD (Y),n
&0C	DSBC A,(X)	&23	ADC A,(U)	&49	AND (X),n	&60	INC Ul
&0D	XOR A,(X)	&24	LD A,Ul	&4A	LD Xl,n	&61	LDI (U),A
&0E	LD (X),A	&25	LD A,(U)	&4B	OR (X),n	&62	DEC Ul
&0F	BIT A,(X)	&26	CP A,Ul	&4C	CP Xh,n	&63	LDD (U),A
&10	SBC A,Yl	&27	CP A,(U)	&4D	BIT (X),n	&64	INC U
&11	SBC A,(Y)	&28	LD Uh,A	&4E	CP Xl,n	&65	LDI A,(U)
&12	ADC A,Yl	&29	AND A,(U)	&4F	ADD (X),n	&66	DEC U
&13	ADC A,(Y)	&2A	LD Ul,A	&50	INC Yl	&67	LDD A,(U)
&14	LD A,Yl	&2B	OR A,(U)	&51	LDI (Y),A	&68	LD Uh,n
&15	LD A,(Y)	&2C	DSBC A,(U)	&52	DEC Yl	&69	AND (U),n
&16	CP A,Yl	&2D	XOR A,(U)	&53	LDD (Y),A	&6A	LD Ul,n

&6B	OR (U),n	&94	LD A,Yh	&AD	XOR A,(N)	&D5	SRA
&6C	CP Uh,n	&95	JRNH,-h	&AE	LD (N),A	&D7	SLD
&6D	BIT (U),n	&96	CP A,Yh	&AF	BIT A,(N)	&D9	SLA
&6E	CP Ul,n	&97	JRH,-n	&B1	SBC A,n	&DB	RLA
&6F	ADD (U),n	&98	????	&B3	ADC A,n	&DD	INC A
&80	SBC A,Xh	&99	JRNZ,-n	&B5	LD A,n	&DE	EVL +d
&81	JRNC,+n	&9A	RET	&B7	CP A,n	&DF	DEC A
&82	ADC A,Xh	&9B	JRZ,-n	&B8	RPV	&E0	ERR
&83	JRC,+n	&9C	DADC A,(Y)	&B9	AND A,n	&E1	SPU
&84	LD A,Xh	&9D	JRNV,-n	&BA	JMP N	&E2	RTS
&85	JRNH,+n	&9E	JR -n	&BB	OR A,n	&E3	RPU
&86	CP A,Xh	&9F	JRV,-n	&BD	XOR A,n	&E4	ERR1
&87	JRH,+n	&A0	SBC A,Uh	&BE	CALL N	&E9	AND (N),n
&88	DJC -n	&A1	SBC A,(N)	&BF	BIT A,n	&EB	OR (N),n
&89	JRNZ,+n	&A2	ADC A,Uh	&C1	CALLNC,(n)	&ED	BIT (N),n
&8A	RETI	&A3	ADC A,(N)	&C3	CALLC,(n)	&EF	ADD (N),n
&8B	JRZ,+n	&A4	LD A,Uh	&C5	CALLNH,(n)	&F1	AEX
&8C	DADC A,(X)	&A5	LD A,(N)	&C7	CALLH,(n)	&F3	RVF
&8D	JRNV,+n	&A6	CP A,Uh	&C9	CALLNZ,(n)	&F5	LD (Y),(X)
&8E	JR +n	&A7	CP A,(N)	&CB	CALLZ,(n)	&F7	CPI A,(X)
&8F	JRV,+n	&A8	SPV	&CD	CALL (n)	&F9	RCF
&90	SBC A,Yh	&A9	AND A,(N)	&CF	CALLV,(n)	&FB	SCF
&91	JRNC,-n	&AA	LD S,N	&D0	ENT n,d	&FD	shift
&92	ADC A,Yh	&AB	OR A,(N)	&D1	RRA	&FE	RESET
&93	JRC,-n	&AC	DADC A,(U)	&D3	SRD		

2EME TABLE
PREFIXE &FD

fonction identique a la 1ere table mais sur page 2:

&01, &03, &05, &07
&09, &0B, &0C, &0D
&0E, &0F, &11, &13
&15, &17, &19, &1B
&1C, &1D, &1E, &1F
&21, &23, &25, &27
&29, &2B, &2C, &2D
&2E, &2F, &49, &4B
&4D, &4F, &59, &5B
&5D, &5F, &69, &6B
&6D, &6F, &8C, &9C
&A1, &A3, &A5, &A7
&A9, &AB, &AC, &AD
&AE, &AF, &E9, &EB
&ED, &EF

Autres fonctions

&0A	POP X	&6A	LD U,X
&18	LD X,Y	&81	EI
&1A	POP Y	&88	PUSH X
&28	LD X,U	&8A	POP A
&2A	POP U	&98	PUSH Y
&40	INC Xh	&A8	PUSH U
&42	DEC Xh	&AA	LD A,F
&48	LD X,S	&B1	HALT
&4C	RBF	&BA	LD A,I
&4E	LD S,X	&BE	DI
&50	INC Yh	&C0	RDP
&52	DEC Yh	&C1	SDP
&58	LD X,P	&C8	PUSH A
&5A	LD Y,X	&CA	ADD X,A
&5E	JMP (X)	&CE	LD I,A
&60	INC Uh	&D3	SRD
&62	DEC Uh	&D7	SLD
		&DA	ADD Y,A
		&EA	ADD U,A
		&EC	LD F,A

Daniel Glazman

CREATE, DATE, DEVICE

Le Modèle 1 possède plusieurs Systèmes d'Exploitation des Disquettes (SED, ou DOS en anglais). Ces différents systèmes ont beaucoup de points communs, particulièrement ceux de la "lignée" TRSDOS-NEWDOS-NEWDOS80-LDOS, qui s'inspirent tous, à quelques détails près, du TRSDOS de Tandy (version initiale 2.1). Nous publierons régulièrement sous forme de fiches pratiques, les commandes de ces différents DOS, classées alphabétiquement. Ces fiches ne prétendent pas remplacer les manuels originaux qu'il est impératif de posséder (il n'était pas envisageable de publier sous cette forme l'équivalent de plusieurs centaines de pages ...).

Leur but est de fournir à l'utilisateur une information rapide d'accès (classement alphabétique), pratique (exemples d'utilisation), comparative (différents DOS) et dans une langue qu'il est censé connaître (français)...

Les DOS les plus répandus ont été retenus pour constituer ces fiches. Ce sont le TRSDOS version 2.3 de Tandy, le NEWDOS80 version 2 d'Apparat et le LDOS version 5.0 de Lobo Drives International. En règle générale, nous nous baserons sur les commandes du TRSDOS, en signalant les différences ou les apports rencontrés dans les autres systèmes d'exploitation.

Conventions d'écriture

Dans la description des commandes, les mots en caractères majuscules désignent les commandes proprement dites, ainsi que les mots-clés associés. Les mots en caractères minuscules ne font pas partie des commandes mais en représentent la forme. Au moment de l'exécution, il faudra leur substituer le ou les mots appropriés. Parmi ceux-ci, nous trouverons par exemple : "nomfich", qui désigne un nom de fichier, "motpasse" qui signifie qu'un mot de passe (password) est attendu à cet endroit. Dans tous les cas, un coup d'oeil sur les exemples donnés devrait lever toute ambiguité.
Enfin, il faut signaler que le NEWDOS80 emploie indifféremment la virgule ou l'espace comme séparateur dans les commandes.

Après l'étude de la commande COPY qui s'est poursuivie sur 4 numéros (TRACE n° 4 à n° 7) nous abordons maintenant la description de nouvelles commandes : CREATE, DATE, DEVICE (la commande DEBUG appelle en fait un utilitaire spécifique, et sera sautée dans cette étude).

CREATE

<u>FORMAT NEWDOS 80 V2.0</u>

CREATE,Specfich1 [,LRL=ln1] [,REC=Compt1] [,ASE=yn] [,ASC=yn]

La commande SED <CREATE> crée un nouveau fichier de nom Specfich1 ou modifie l'état d'un fichier existant Specfich1.
Cette commande permet à l'utilisateur de préallouer l'espace disque nécessaire à un fichier.

Les paramètre utilisables sont les suivants :

LRL= n : Spécifie la longueur de chaque enregistrement du fichier 1 $\leqslant$ n $\leqslant$ 256
Valeur par défaut = 256.

REC=Compt1 : Spécifie le nombre d'enregistrements initialement alloués au fichier
Valeur par défaut = 0.

ASE=yn : Ce paramètre indique si le SED doit allouer automatiquement après la commande CREATE un espace supplémentaire disque pour le fichier
ASE=Y autorise cette allocation, ASE=N l'interdit.
La valeur par défaut est ASE=Y.

ASC=yn : Ce paramètre indique si la fonction de fermeture de fichier du SED permet de désallouer automatiquement l'espace disque en excédent.
ASC=Y autorise cette désallocation
ASC=N l'interdit.
La valeur par défaut est ASC=Y.

<u>FORMAT LDOS V 5.1</u>

CREATE Specfich (LRL=aaa,REC=bbbb,SIZE=cccc)

Même rôle que sous NEWDOS 80 V2.0
Paramètres possibles :

LRL=aaa : longueur de l'enregistrement logique
1 $\leqslant$ LRL $\leqslant$ 256
Valeur par défaut 256

REC=bbbb : nombre d'enregistrements de longueur LRL=aaa à allouer au fichier.

SIZE=cccc : espace disque en kilo-octets que le fichier peut occuper. SIZE ne peut être spécifier si LRL et REC le sont.
Abréviations : LRL=L, RECH=R, SIZE=S.

DATE

FORMAT NEWDOS 80 V2.0, LDOS V5.1, TRSDOS

DATE [mm/jj/aa]

Etablissement de la date courante suivant le format :
- mm=mois, jj=jour, aa=année

avec 01 ⩽ mm ⩽ 12 , 01 ⩽ jj ⩽ 31 et 00 ⩽ aa ⩽ 99
Si aucun paramètre n'est spécifié la comma fiche la date courante de l'horloge en temps réel sous le format mm/jj/aa.

Remarque 1 : Sous LDOS ou NEWDOS80, le message DATE? ou DATE apparaît à la mise sous tension, l'utilisateur doit fournir une réponse suivant le format mm/jj/aa, la plage autorisée de réponse est comprise entre 01/01/80 et 12/31/87 (LDOS).

LDOS : Une fois là réponse donnée le SED affiche l'abréviation du jour de la semaine, le nom du mois (abréviation) et le n° du jour dans le mois et l'année complète (1980 à 1987). Le message de demande de DATE peut être supprimé avec l'option SYSTEM (DATE=NO).

Remarque 2 : Sous NEWDOS 80 V2.0 la dévalidation correspondante est possible avec les options de la commande SYSTEM (AY=N et AZ=N).

DEVICE (n'existe pas sous NEWDOS 80)

Affichage de tous les dispositifs logiques utilisés par le système avec les abréviations
KI = clavier (KEYBOARD INPUT)
DO = écran (DISPLAY OUTPUT)
PR = imprimante (PRINTER)

(autres dispositifs logiques sous LDOS : * JL, * SI, * SO, * UD).
LDOS fournit également les caractéristiques des disquettes utilisées dans les unités disponibles (nombre de faces, densité, localisation de la piste répertoire, nombre de pistes, temps d'accès piste) sous le format :

aabb c dddddd ee fffffff gggg iiiiiiiii jjjjj

avec : aa = n° de DRIVE
bb = état de protection de la disquette (option SYSTEM : DRIVE = WP)
cc = taille du "floppy" ou "hard" disque 5 pouces ou 8 pouces
dd = type de DRIVE : souple ou rigide
ee = pour les disques souples : emplacement physique du DRIVE sur son câble (1,2,4,8) pour les disques durs : n° de la tête de départ
ff = nombre de cylindres (pistes)
gg = densité de stockage (DDEN ou SDEN)
hh = nombre de faces (1 ou 2)
ii = temps d'accès piste à piste en ms pour l'unité de disque considérée
jj = temps d'attente entre le démarrage du moteur et l'accès possible à la disquette.

Index des mots-clés

La liste des mots-clés figurant ci-dessous est destinée à être exploitée par le programme de gestion des articles décrit dans le numéro 1 de TRACE (pages 58 à 64). Le format est le suivant:

Nom abrégé de l'article, numéro de page : 1 à 4 mots-clés.

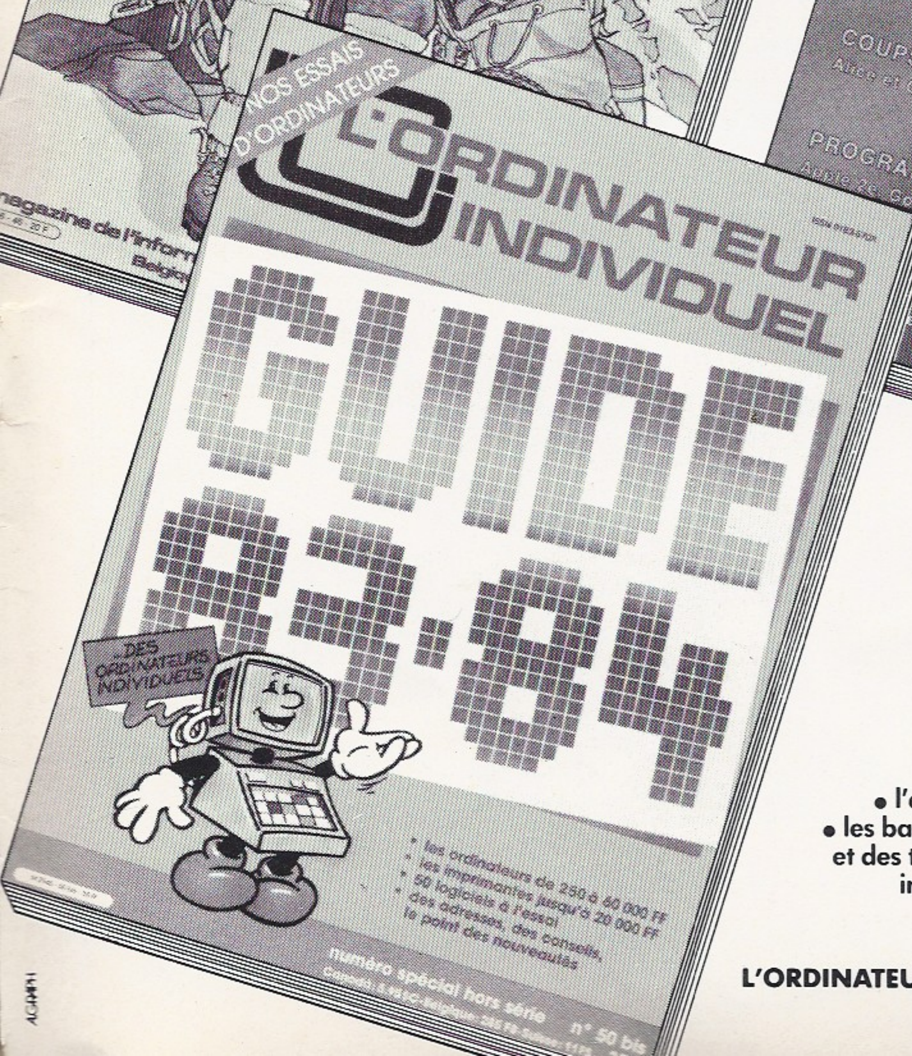
En prenant l'habitude d'entrer cette liste à chaque numéro de TRACE, vous constituerez ainsi une petite base de données vous permettant de trouver plus efficacement et plus rapidement l'article que vous cherchez. Bien entendu, cette liste est donnée à titre indicatif, et vous pourrez toujours en modifier ou en compléter le contenu, afin de l'adapter à vos besoins ; mais avant toute chose, vous devez vous imposer une certaine normalisation des mots-clés (en principe, ils sont limités à 12 caractères) PROGRAMME par exemple, indiquera toujours qu'un listage de programme est donné dans l'article. Pour cela aussi, faites-nous part de vos idées!

Imprimerie CEPI, 75010 Paris. Imprimé en France. Directeur de la publication Jean-Pierre Nizard
Dépôt légal : Avril 1984

pour mieux choisir votre ordinateur et pour mieux l'utiliser :







lisez

L'ORDINATEUR INDIVIDUEL

Vous y trouverez :

• l'actualité et les tendances de l'informatique individuelle • les bancs d'essais des principaux matériels • des panoramas et des tests comparatifs • le point des grandes manifestations internationales • des articles d'initiation • des synthèses • des programmes • des interviews "exemplaires" • des conseils • des idées • des astuces

L'ORDINATEUR INDIVIDUEL, chez votre marchand de journaux